

UNIVERSIDADE TÉCNICA DE LISBOA

INSTITUTO SUPERIOR TÉCNICO



Classificação de Poemas e Sugestão das Palavras Finais dos Versos

Paulo Alexandre Medeiros de Araújo
(Licenciado)

Dissertação para obtenção do Grau de Mestre em
Engenharia Electrotécnica e de Computadores

Orientador Científico: Professor Doutor Nuno João Neves Mamede

JÚRI

Presidente: Professora Doutora Isabel Maria Martins Trancoso

Vogais: Professor Doutor Paulo Miguel Torres Duarte Quaresma
Professor Doutor Nuno João Neves Mamede
Professora Doutora Helena Sofia Andrade Nunes Pereira Pinto

NOVEMBRO 2004

À Rita.

RESUMO

Esta dissertação apresenta uma arquitectura que permite realizar a classificação automática de poemas e sugere as palavras finais dos versos.

A classificação dos poemas tem como base os conceitos da poética portuguesa, que se dividem em conceitos estruturais e em regras de classificação. A classificação de poemas é determinista e não necessita de exemplos de poemas para ser realizada.

A sugestão das palavras finais dos versos é realizada com diferentes critérios de selecção e ordenação das palavras a sugerir. O primeiro critério baseia-se nas restrições estruturais dos poemas. O segundo critério baseia-se na categoria gramatical das palavras a sugerir. O terceiro critério baseia-se na utilização de modelos estatísticos de língua obtidos a partir de corpora de texto.

São classificados poemas realizados por crianças em idade escolar e também algumas estrofes de autores conhecidos, como António Aleixo e Camões. São comparados exemplos de sugestão das palavras finais dos versos, utilizando os vários critérios de selecção e ordenação enumerados.

Das várias utilizações possíveis para o sistema, destaca-se a sua utilização como ferramenta didáctica de apoio ao estudo de poesia nas escolas, como auxiliar de escrita de poesia e como auxiliar de leitura em voz alta de poemas a partir da informação de classificação respectiva.

Palavras Chave

Inteligência Artificial, Processamento de Língua Natural, Modelos Estatísticos de Língua, Poesia, Rima.

ABSTRACT

This dissertation presents a structure that allows automatic classification of poems and final word suggestion to the poem's verses.

The Portuguese poetic concepts are the basis for poem classification, which are divided in structural concepts and classification rules. The poem classification is determinist and doesn't need any example of poems to be produced.

The final word suggestions of the verses are made by different selection and order criteria of the words to suggest. The first criterion is based on the structural restrictions of the poems. The second criterion is based in grammatical category of the suggested words. The third criterion is based on statistics language models that can be obtained from a text corpus.

Poems made by children in school age and some strophes made by known poets like António Aleixo and Camões are classified. Final word suggestions are generated and compared to the previously enumerated selection and order criteria.

From the various possible utilizations for this system, it can be use as a didactic tool to the poetry study in schools, poetry help writer, and poetry reading out loud aid.

Key Words

Artificial Intelligence, Natural Language Processing, Statistic Language Models, Poetry, Rhyme.

AGRADECIMENTOS

Agradeço ao meu orientador, Professor Nuno Mamede, pelas suas ideias e saber, pela sua orientação e correcção, que permitiram a concretização desta dissertação.

Ao grupo de investigação do *INESC*, L^2F, do qual faço parte, agradeço todo o seu apoio e contributos, as suas ideias e opiniões, e os trabalhos facultados que integram esta dissertação como são exemplo o *Leia*, o *Smorph*, o *Pasmo* e a *Susana*. Em particular, ao Professor Luís Caldas, à Professora Isabel Trancoso, à Luísa Coheur, ao Hugo Meinedo, ao Fernando Batista e à Joana Paulo, agradeço todo o apoio e ajuda que permitiram quebrar a barreira inicial na utilização dos vários módulos de software, novos para mim, e pela disponibilidade em me ajudar a compreendê-los.

Agradeço à Professora Ana Paiva e aos alunos Ana Pacheco, Joana Paulo e Nuno Ferreira pela disponibilização do projecto de introdução aos agentes, que me deu algumas ideias no início da realização deste trabalho.

Aos meus colegas do *ISEL* Walter Vieira, Helder Pita, Luís Morgado, Paulo Trigo, António Teófilo, e Porfírio Filipe, agradeço a sua paciência para me ouvir e os seus contributos e opiniões. Ao meu amigo Porfírio Filipe, agradeço as acesas discussões que transformaram algumas das ideias em concretizações.

Agradeço à Rita, minha esposa, que me acompanha e ajuda em todos os dias da minha vida.

À minha família, que desde a infância me acompanha e me incentiva na caminhada que me permitiu chegar aqui, agradeço o seu apoio incondicional.

Por último, mas não menos importante, gostaria de agradecer ao *PRODEP* pela dispensa de serviço concedida que muito contribuiu para que este trabalho chegasse a bom porto.

ÍNDICE

LISTA DE FIGURAS

LISTA DE TABELAS

CONVENÇÕES TIPOGRÁFICAS

As convenções tipográficas utilizadas nesta tese foram as seguintes:

i) usa-se a fonte "Times New Roman" para texto normal

Exemplo: Esta proposta pretende abordar dois problemas distintos;

ii) usam-se parêntesis rectos [ ] para referências bibliográficas

Exemplo: [Araújo & Mamede, 2002];

iii) usa-se texto em "*Itálico*" entre aspas para frases citadas

Exemplo: "*Considera-se poema toda composição literária de índole poética*";

iv) usa-se texto em *Itálico* para siglas e abreviaturas

Exemplo: sistema *LuCas*;

v) usam-se plicas ‘’ para destacar pontos relevantes

Exemplo: as palavras ‘Vistas’ e ‘Largas’;

vi) usa-se a fonte "`Courrier New`" nos exemplos de código

Exemplos: `if(txtLinha.compareTo("") != 0)`

1 INTRODUÇÃO

1.1 Motivação

Hoje em dia, é cada vez maior a utilização de ferramentas de apoio à escrita, como são exemplo os correctores ortográficos, que ajudam na edição e correcção de documentos. Estas ferramentas são cada vez mais necessárias e a dependência em relação a elas torna-se cada vez maior. Cada vez menos se concebe que um documento seja enviado com erros ortográficos, pelo menos, aqueles que podem ser detectados por correctores ortográficos.

A integração destas ferramentas nos editores de texto, quer através de opções de menu quer através de botões nas barras de ferramentas, torna a sua utilização bastante simples. As funções mais comuns que são disponibilizadas nos editores mais comuns estão relacionadas com a ortografia e com a gramática. Se se pretender funções específicas para tipos concretos de textos literários, como são exemplo os textos poéticos, já as ferramentas mais comuns podem não implementar essas funções.

Os textos literários podem ser divididos em textos em prosa e textos em verso. As aplicações de edição de texto permitem editar ambos os tipos de texto devido à suas funções serem genéricas e permitirem ser aplicadas aos diferentes tipos de texto.

"*A poesia distingue-se da prosa, não só pelo seu aspecto formal, que facilmente identificamos, mas ainda e sobretudo pela cadência e musicalidade, pela inspiração ardente, pelo arrojo das imagens, pela beleza da expressão e pelo encanto secreto que devem existir em todos os versos dignos deste nome*" [Areal, 2000]. Esta citação, salienta as diferenças entre os textos em prosa e os textos em verso.

Esta diferença pode ser dividida em duas partes: por um lado a cadência e musicalidade que estão relacionadas com a estrutura e a sonoridade das palavras que compõem os textos em verso; e

por outro a inspiração, as imagens e a beleza que estão relacionadas com factores de interpretação mais subjectivos e que dependem da sensibilidade de cada pessoa.

Da definição apresentada pode-se concluir que, se se pretender um sistema automático para analisar ou manipular textos em verso, terá que se ter em conta a cadência e a musicalidade. Existem poucas ferramentas ou utilitários que permitem manipular textos poéticos e reconhecer os requisitos específicos que são característicos da poesia. Esta escassez é agravada quando se pretende editar poesias escritas na língua portuguesa.

A edição de textos poéticos pode ser simplificada com uma ferramenta que realize operações de análise e classificação de poemas e, inclusivamente, permita realizar a sugestão de palavras. Esta ajuda é mais significativa nas palavras finais dos versos, nos poemas em que existe rima. Uma ferramenta com estas características permite:

- Apoiar as pessoas que iniciam o estudo de poesia;
- Incentivar o gosto pela poesia;
- Ajudar os poetas a realizar poesia;
- Ajudar a compreender a estrutura dos poemas para melhorar a sua leitura em voz alta.

Estes são só alguns exemplos de possíveis utilizações para um sistema com as características apresentadas, pois se se considerarem as aplicações lúdicas então é possível realizar jogos de palavras baseados em textos poéticos que permitem ensinar poesia de uma forma divertida. No entanto, não foi este o âmbito do trabalho. Outra razão que motivou a realização desta proposta foi o desejo de aprofundar o conhecimento na área do Processamento da Língua Natural.

Tendo como base as funcionalidades anteriormente descritas, e os cenários de possíveis aplicações enumerados, foram definidos os seguintes objectivos principais:

- Realizar a classificação de diferentes tipos de poesia;
- Apresentar de forma simples e clara o resultado da classificação dos poemas;
- Ter como base, para a classificação de poemas, os conceitos da poética portuguesa sem necessitar, à partida, de exemplos de poemas que condicionem a forma como é realizada a classificação;
- Sugerir as palavras finais dos versos;
- Utilizar diferentes critérios de selecção das palavras finais a sugerir.

Estes objectivos podem ser agrupados sob a perspectiva de dois problemas que são distintos. Por um lado, os três primeiros objectivos relacionados com a classificação de poemas e, por outro

lado, os dois últimos objectivos que estão relacionados com a sugestão de palavras. Embora possam ser analisados de forma distinta eles estão relacionados entre si.

Para o caso da classificação de poemas destaca-se, no 3º objectivo, o facto de se pretender que o resultado de classificação não seja influenciado por modelos de poemas previamente fornecidos ao sistema para daí inferir as regras de classificação. Em vez disso foi realizada uma pesquisa bibliográfica com o objectivo de encontrar as definições dos conceitos e regras da poética portuguesa e a classificação dos poemas foi implementada com base nessas definições.

No caso da sugestão de palavras são usadas técnicas com diferentes critérios para seleccionar e ordenar as palavras a sugerir. Um dos critérios de selecção de palavras tem em conta as restrições estruturais do poema, e estas fazem parte da informação de classificação dos poemas, razão pela qual, os dois problemas não são independentes.

1.2 Enquadramento

Esta proposta enquadra-se na área do Processamento da Língua Natural, pois são utilizadas técnicas de Processamento da Língua Natural para realizar a classificação dos poemas e a sugestão das palavras finais dos versos.

No caso da classificação de poemas, o resultado da pesquisa bibliográfica efectuada permitiu implementar, a partir das definições de aspectos formais, nomeadamente os conceitos e as regras da poética portuguesa, um conjunto de conceitos e regras para o sistema. Existem dois aspectos que são fundamentais, um corresponde à rima das palavras finais dos versos e o outro ao número de sílabas das palavras que formam os versos. Para contemplar estes dois aspectos é necessário um mecanismo de verificação de rima das palavras e um mecanismo de decomposição das palavras em sílabas.

Para o caso da sugestão das palavras finais dos versos foram implementadas várias hipóteses de critérios de escolha e ordenação das palavras. A escolha das palavras a sugerir é condicionada pela estrutura do poema que está a ser construído e, naturalmente, pela rima e pelo número de sílabas dos versos e das palavras anteriores, caso existam. Para as palavras que se encontram nestas condições é realizada uma ordenação que permite eleger as palavras que melhor se adequam à posição do poema onde se pretende realizar a sugestão.

No âmbito deste trabalho, os critérios que foram escolhidos para realizar a filtragem das palavras, por um lado, tiram partido da análise sintáctica das frases que constituem o poema com o objectivo de seleccionar as categorias possíveis para a palavra que deve ser sugerida, e por outro

lado, são usados modelos estatísticos de língua, que permitem realizar uma ordenação das palavras segundo um critério baseado no seu valor estatístico de ocorrência.

Foram realizadas pesquisas na Internet relacionadas com poesia com o objectivo de encontrar sistemas que permitam lidar com poesia. Encontraram-se dois tipos de *Sites*. No primeiro tipo é possível ler, comentar e discutir poesia de vários autores. O segundo tipo disponibiliza sistemas que permitem lidar com poesia, que vão desde os jogos de palavras até à geração automática de poesia.

As pesquisas de *Sites* efectuadas abrangeram não só a língua portuguesa como outras línguas e dos vários *Sites* encontrados, a maioria é sobretudo para a língua inglesa. Alguns dos *Sites* encontrados para a língua portuguesa são em Português do Brasil.

Começando pelos *Sites* do primeiro tipo, para a língua portuguesa destacam-se o [Projecto Vercial] e [Geração Poesia]. Em ambos os *Sites* é possível encontrar um vasto número de poemas de autores Portugueses como são exemplo Luís de Camões e Fernando Pessoa.

O objectivo destes *Sites* é juntar textos literários de uma grande diversidade de autores e divulgar a literatura para motivar as pessoas para a leitura. Alguns fazem ainda concursos de poesia onde existem processos de votação dos poemas e permitem que os leitores façam comentários sobre os poemas para serem partilhados por outras pessoas.

Dos vários *Sites* estrangeiros encontrados destacam-se, para a língua inglesa, dois *Sites* [Poetry Library] e [E-Poetry]. Em ambos é possível encontrar muitos autores de diferentes nacionalidades e muitos poemas para ler. No primeiro *Site* são disponibilizados mecanismos que permitem dar apoio aos poetas que desejam publicar os seus trabalhos. No segundo *Site* são inclusivamente anunciados eventos com o objectivo de divulgar sistemas vocacionados para poesia.

Passando agora para os sistemas que permitem a manipulação de texto em que os poemas estão contemplados,. existe uma grande variedade de sistemas que vão desde os simples jogos de palavras, em que o objectivo é construir um poema, até aos sistemas mais complexos que geram modelos de representação de poesia e permitem gerar poemas automaticamente.

Como exemplos destacam-se alguns dos vários sistemas encontrados, sendo a ordem de apresentação dos mais simples para os mais complexos:

- O *Chaos Poetry Generator* [Productions, 1997] consiste num gerador aleatório de caracteres, onde é possível controlar os grupos de caracteres gerados e, assim, criar poesia;

- O *Electric Poet* [Frykholm, 1996] permite criar um trabalho literário a partir de um texto normal que serve de molde. Este sistema está mais vocacionado para a poesia abstracta;
- O *Mc Poet* [Westbury, 1997] é um conjunto de ferramentas para manipulação de textos poéticos. Estas ferramentas têm um motor de geração de texto baseado num sistema de regras que usa uma linguagem simples para permitir a configuração por parte do utilizador;
- O *Dada Poem Generator* [Chachanashvili, 1991] permite gerar automaticamente poemas sem sentido, tendo como base um dicionário e definições sintácticas;
- *O Ray Kurzweil Cybernetic Poet* [Kurzweil, 1999] é, das ferramentas analisadas, a mais completa. Esta ferramenta faz a aquisição de um conjunto de poemas e cria um modelo de língua que representa esse mesmo conjunto de poemas. A partir do modelo criado permite gerar poemas. Para além deste sistema, a empresa Kurzweil CyberArt Technologies dedica-se à criação de sistemas capazes de produzir arte e inclusivamente disponibilizou um sistema de protecção de ecrã capaz de gerar pinturas originais, o AARON [Cohen, 2001].

Existem dicionários de rima vocacionados para a realização de poemas, e existem tanto em formato impresso em papel como em formato electrónico. A função de um dicionário de rimas é permitir procurar palavras que rimam e a sua utilidade é facilitar a procura de uma palavra para uma determinada posição do poema. O utilizador apenas tem de fornecer as últimas letras da palavra que pretende encontrar, e o sistema fornece as palavras do dicionário com as mesmas letras no fim da palavra.

As versões impressas em papel estão organizadas alfabeticamente pelas letras finais das palavras, ou seja, as palavras são ordenadas alfabeticamente do fim da palavra para o início. Isto permite juntar as palavras com igual terminação em termos das letra finais. Esta abordagem permite encontrar mais facilmente as palavras com igual terminação e assim procurar palavras que rimem. No entanto, existem palavras que rimam e que não têm exactamente a mesma terminação em termos de letras.

Os dicionários em formato electrónico podem ser usados de duas formas: ou através de *Sites* onde o utilizador usa as funções aí disponíveis; ou através de aplicações que funcionam de forma autónoma ou integradas num editor de texto. As funções disponibilizadas pelos dicionários em formato electrónico são as mesmas funções possíveis de realizar na versão em papel, com a vantagem de quando estão integradas nas ferramentas de edição de texto tornam a pesquisa das palavras mais simples e cómoda.

Um dos dicionários de rimas em formato electrónico que foi testado designa-se por *Dicionário de Rimas Poéticas* [Pretor, 2000]. Este dicionário é apresentado como um "*adminículo ao poeta, sem nunca ter a pretensão de substituí-lo, porque insubstituível a criação do espírito; apenas auxiliará na busca da rima mais apropriada dispensando a fastidiosa consulta a dicionários convencionais*" [Pretor, 2000]. Esta aplicação é um exemplo do funcionamento que foi descrito para os dicionários de rima em formato electrónico na forma de aplicação independente. A forma como são realizadas as pesquisas das palavras é com base nas últimas letras das palavras.

O sistema apresentado nesta dissertação também sugere as palavras finais dos versos sendo um dos critérios de selecção a rima das palavras. Esta funcionalidade é equivalente à utilização de um dicionário de rimas anteriormente descrito. No entanto, a abordagem não está baseada na terminação em termos de letras das palavras, mas sim na sonoridade das palavras. Assim, quando se pretende obter palavras que rimam com uma determinada palavra dada, obtêm-se as palavras que em termos sonoros são semelhantes a essa palavra (consonantes).

A diferença da abordagem seguida, baseada na sonoridade das palavras, tem a vantagem de, dada uma palavra, eliminar as palavras com igual terminação em termos de letras mas que não são consonantes e juntar as palavras que têm terminações diferentes em termos de letras mas que são consonantes.

Neste trabalho, a forma como o utilizador indica a palavra para a qual pretende obter as palavras que rimam teve de ser diferente da dos sistemas analisados. Enquanto que nas aplicações testadas o utilizador fornece apenas as letras finais da palavra, no sistema implementado tem de ser fornecida a palavra completa. O problema reside no facto das letras finais sem o contexto da palavra não determinarem quais as palavras que são consonantes a esse grupo de letras.

Esta abordagem facilita o mecanismo automático de pesquisa de palavras que rimam, necessário na classificação de poemas, pois é possível saber se duas palavras rimam sem ter de decompor as palavras em letras para fornecer as respectivas terminações ao sistema.

Neste trabalho, e tendo como referência os objectivos já enumerados, existe um conjunto de considerações que foram tidas em conta:

- A classificação de poemas deve ser possível de realizar tendo apenas como base as definições dos conceitos e regras da poética portuguesa;
- A classificação automática de poemas deve ser o mais genérica possível por forma a cobrir o maior número de poemas;
- A sugestão de palavras deve ser feita tendo em conta diferentes critérios de selecção de palavras para permitir uma escolha mais flexível.

Assim, para se conseguirem atingir os objectivos enumerados, existe um conjunto de funcionalidades que são necessárias:

- Transcrição fonética das palavras;
- Divisão silábica das palavras;
- Análise morfológica das palavras;
- Análise sintáctica das frases.

Existem, no âmbito do grupo de investigação do L^2F, ferramentas que implementam estas funcionalidades. Neste trabalho considera-se aplicação externa aquela que implementa uma ou várias funcionalidades descritas. O sistema interage com estas aplicações de forma a adquirir a informação necessária quer para a classificação de poemas quer para sugestão de palavras.

1.3 Organização da Dissertação

No capítulo 2 começa-se por fazer uma introdução à modelação de língua destacando alguns aspectos relevantes relacionados com a área da linguística e da fala. São apresentados os modelos formais fazendo referencia à classificação dos vários tipos de gramáticas segundo Chomsky. Seguidamente é feito um enquadramento dos modelos estatísticos de língua que são usados no contexto deste trabalho, como critério de decisão na escolha das palavras. É ainda feita uma referência à ferramenta que permite gerar os modelos de língua e são apresentados exemplos sobre o modelo de língua utilizado no âmbito deste trabalho.

No capítulo 3 são introduzidas quatro definições base: a acentuação das palavras; a transcrição fonética; a verificação da rima e a divisão das palavras em silábicas gramaticais e em sílabas métricas. Discute-se a utilização das categorias morfológicas das palavras utilizadas no processo de sugestão, como meio de selecção das palavras segundo a sua categoria. Descrevem-se os conceitos estruturais da poética portuguesa, que servem de base à classificação de poemas, sendo indicadas as opções de implementação que foram tomadas. Enumeram-se as regras de classificação dos poemas e as respectivas restrições de implementação. É também apresentado o que se entende por tipos de poesia e são dados vários exemplos de poesia aceites pelo sistema.

No capítulo 4 são descritos os aspectos que foram tidos em conta na construção do demonstrador, sistema *LuCas* fazendo referência à motivação que levou à corrente implementação. Descreve-se em detalhe a arquitectura do sistema e os vários módulos que o compõem e as suas respectivas interfaces. Descrevem-se os processos de coordenação do sistema e apresentam-se em UML os diagramas de estado e de actividade correspondentes à forma como é realizada a

classificação dos poemas e a sugestão das palavras finais dos versos. São discutidas várias abordagens para selecção e ordenação das palavras a sugerir. Apresenta-se a arquitectura da base de dados do sistema e os seus detalhes de implementação. São apresentados os componentes que armazenam a informação obtida pelas aplicações externas e alguns resumos estatísticos de ocorrência de terminações de palavras e de número de sílabas dos modelos estatísticos de língua. Em seguida descrevem-se alguns exemplos que ilustram o funcionamento do sistema na classificação de poemas e na sugestão das palavras finais dos versos.

No capítulo 5 começa-se por apresentar a interface do sistema *LuCas* implementado e o resumo dos comandos disponíveis na sua interface. São apresentados resultados de avaliação do classificador de poemas em termos de tempos de resposta do sistema. Em seguida apresentam-se os resultados de avaliação do processo de sugestão de palavras sendo feito uma comparação entre as várias abordagens possíveis na sugestão das palavras, nomeadamente as restrições estruturais do poema e a utilização dos modelos estatísticos de língua.

No capítulo 6 são feitas as conclusões finais desta dissertação. Começa-se por resumir o estado actual do sistema e as suas limitações. São focadas as perspectivas futuras e discutidas as vantagens e desvantagens da integração do sistema numa plataforma *Galaxy* e ainda os possíveis caminhos a seguir por forma a melhorar quer aspectos de classificação quer de predição de palavras.

2 MODELAÇÃO DE LÍNGUA

2.1 Introdução

A modelação de língua é uma área de investigação vasta e activa quer na comunidade da fala como na comunidade da linguística. Estas duas comunidades têm abordagens distintas ao problema, que levam aos modelos probabilísticos de língua e às teorias formais de língua.

Antes de Chomsky eram aplicadas aproximações distribuídas que usavam restrições de contexto na modelação de língua [Hutchens, 1995]. Chomsky apresentou argumentos tais como métodos probabilísticos e introduziu uma aproximação formal baseada em gramáticas bem definidas.

Na área da linguística foram desenvolvidas ferramentas para realizar tarefas como a análise sintáctica e a análise semântica de frases. A complexidade associada a essas ferramentas, de um modo geral, é polinomial e está dependente do comprimento das frases [Huang et al., 2001].

Existem dois aspectos fundamentais nos modelos formais de língua: a gramática e os algoritmos de análise. A gramática é uma especificação formal das estruturas possíveis para a língua. As técnicas de análise são métodos para analisar frases e verificar se as estruturas são compatíveis com a gramática. Este processo requer uma grande quantidade de textos analisados gramaticalmente por processo manual para conseguir avaliar estas ferramentas, na ordem das dezenas ou mesmo centenas de milhão de palavras.

Por outro lado na área da fala foram desenvolvidas ferramentas para prever a próxima palavra com base no que já foi dito. Um dos objectivos associados a esta técnica é melhorar os resultados do reconhecimento da fala.

As relações de probabilidade entre sequências de palavras podem ser modeladas a partir de corpus de texto com os chamados modelos probabilísticos de língua, como são exemplo os *Ngramas,* e em oposição à utilização de extensas gramáticas formais. Também neste caso é

importante existir essa grande quantidade de textos para que inclua o vocabulário que se pretende analisar.

Nenhuma das aproximações é completamente bem sucedida pois, se por um lado, as gramáticas formais não são suficientemente robustas e necessitam de grande esforço para as adaptar de um domínio para outro, por outro lado, a falta de estrutura e compreensão dos modelos probabilísticos retiram-lhes a facilidade em escolher as palavras certas para guiar o reconhecimento da fala.

2.2 Modelos Formais de Língua

Na teoria formal de língua, a construção da gramática tem de considerar a generalidade, a selectividade e a compreensão. A generalidade está associada ao conjunto de frases analisadas correctamente e a selectividade está associada ao conjunto de frases que são identificadas como problemáticas. A compreensão está associada à simplicidade da gramática e é importante para permitir que a gramática seja mantida.

Segundo Chomsky [Chomsky, 1965], as gramáticas podem-se dividir por 4 tipos: as gramáticas de estrutura de frase, as gramáticas sensíveis ao contexto, as gramáticas independentes do contexto e as gramáticas regulares. Estas gramáticas estão estruturadas hierarquicamente pelos quatro tipos de autómatos que aceitam as linguagens produzidas pelos quatro tipos de gramáticas: são respectivamente máquinas de Turing, autómatos lineares, autómatos *Push Down* e autómatos de estados finitos.

As gramáticas de estrutura de frase apenas têm uma restrição na forma das suas regras ($\alpha \rightarrow \beta$), a parte esquerda nunca pode ser nula. Este primeiro tipo caracteriza as linguagens recursivamente enumeradas, ou seja, aquelas que são listadas por uma máquina de Turing. As gramáticas sensíveis ao contexto têm regras que sobrepõem um símbolo não terminal num determinado contexto por um conjunto de símbolos não nulos. As gramáticas independentes do contexto permitem que as regras sobreponham qualquer símbolo não terminal por símbolos terminais ou não terminais. As gramáticas regulares são equivalentes às expressões regulares, ou seja, uma linguagem regular é caracterizada por uma expressão regular ou por uma gramática regular. As gramáticas regulares podem ser lineares à direita ou lineares à esquerda. As regras lineares à direita têm um único símbolo não terminal à esquerda e no máximo um símbolo não terminal à direita. No exemplo(ex: $A \rightarrow wB$) em que w é um símbolo terminal e A e B são símbolos não terminais [Huang et al., 2001].

Desde que foi introduzida a noção de gramáticas independentes do contexto que surgiu uma vasta literatura sobre algoritmos de análise. Muitas delas com o objectivo de analisar linguagens de programação que não são ambíguas, que não é o caso da linguagem falada.

Existem duas aproximações distintas, por um lado temos os algoritmos descendentes e por outro lado temos os algoritmos ascendentes. Os algoritmos descendentes começam por um símbolo que representa a frase e vão substituindo os símbolos de diferentes formas até formar uma árvore que representa a frase ou até esgotar todas as hipóteses possíveis. Os algoritmos ascendentes começam pelas palavras que compõem a frase e vão substituindo por símbolos e formando uma árvore até ter apenas um único símbolo que representa a frase.

Early [Early, 1970] implementou um algoritmo descendente com reconhecimento ascendente que consegue ir buscar o que têm de melhor ambos os algoritmos ascendente e descendente.

2.3 Modelos Estatísticos de Língua

Os modelos estatísticos de língua estimam a distribuição probabilística de vários fenómenos de língua natural [Rosenfeld, 2000]. Ironicamente, o sucesso das técnicas de modelos estatísticos de língua usam muito pouco do que a linguagem é. Os modelos de língua de *Ngramas* não tiram partido do facto de se estar a modelar uma Língua Natural. Poderia ser uma sequência de símbolos arbitrária sem uma estrutura profunda, intenção ou pensamento por trás [Rosenfeld, 2000].

Os modelos de língua, necessários à sugestão de palavras, podem ser inferidos a partir de um corpus de texto, e sem necessitar de qualquer outra informação adicional à partida, podendo depois ser usados por um algoritmo na predição de palavras [Hutchens, 1995].

O objectivo principal dos modelos probabilísticos de língua é providenciar informação estatística para que as sequências de palavras mais prováveis tenham maior probabilidade do que as menos prováveis. Deste modo é possível melhorar a precisão e reduzir o espaço de procura no reconhecimento da fala [Huang et al., 2001].

Neste trabalho, a utilização dos modelos de língua permite seleccionar, para uma determinada posição do texto, um conjunto de palavras e estabelecer uma ordem nas palavras, tendo como critério de ordenação a frequência de ocorrência. Para cada um dos modelos de *Ngramas*, seleccionam-se sempre as palavras com maior frequência.

Se se considerar em W sequências de palavras então pode-se assumir que nos modelos de língua de *Ngramas*, P(W) reflecte a distribuição de probabilidade de ocorrência das sequências de palavras W.

Por exemplo, num modelo que descreve a língua falada, pode-se ter P(olá)=0.01, pois provavelmente uma em cada cem palavras é 'olá'. Por outro lado também se pode encontrar P(fazer gato quadro)=0, pois é extremamente improvável alguém proferir esta frase.

P(W) pode ser decomposto em:

$$\begin{aligned} P(W) &= P(w_1, w_2, ..., w_n) \\ &= P(w_1)\, P(w_2 \mid w_1)\, P(w_3 \mid w_1, w_2) \ldots P(w_n \mid w_1, w_2, ..., w_{n-1}) \\ &= \prod_{i=1}^{n} P(w_i \mid w_1, w_2, ..., w_{i-1}) \end{aligned}$$

Onde $P(w_i \mid w_1, w_2, ..., w_{i-1})$ é a probabilidade de ocorrência w_i dada a sequência de palavras $w_1, w_2, ..., w_{i-1}$ aparecer previamente.

Como se pode constatar, as palavras anteriores condicionam as palavras que são sugeridas e também o número de palavras anteriores existentes condicionam quais os modelos de *Ngramas* a utilizar. Para um vocabulário de dimensão v existem v^{i-1} diferentes hipóteses e para especificar completamente $P(w_i \mid w_1, w_2, ..., w_{i-1})$, têm de ser estimados v^i valores [Huang et al., 2001]. Mesmo para valores moderados de i é impossível estimar todos os valores pois a maior parte deles só ocorrem uma vez ou muito poucas vezes. A solução prática é restringir o número de palavras anteriores, levando aos modelos de *Ngramas*:

- Os modelos de língua *unigramas* $P(w_i)$ apenas consideram a frequência de ocorrência da palavra sem considerar as palavras anteriores;
- Os modelos de língua *bigramas* $P(w_i \mid w_{i-1})$ consideram apenas a palavra anterior no cálculo da frequência de ocorrência;
- Os modelos de língua *trigramas* $P(w_i \mid w_{i-2}, w_{i-1})$, consideram as duas palavras anteriores no cálculo da frequência de ocorrência.

Desta forma, os modelos estatísticos de língua, através dos *Ngramas*, permitem representar um determinado corpus de texto e realizar a sugestão das palavras com base na frequência de ocorrência.

Um dos problemas existentes nos modelos de *Ngramas* deve-se ao facto de, mesmo utilizando corpus de treino grandes, existirem *Ngramas* que são válidos em termos de linguagem

mas não existem no corpus de treino e por isso a sua ocorrência é nula. Este problema é tanto maior quanto maior for o número de palavras anterior que se quiser considerar.

Existem técnicas que permitem lidar com este problema e permitem atribuir frequências diferentes de 0 aos *Ngramas* que não registam ocorrências. Esta técnica é designada por alisamento.

O primeiro método para realizar o alisamento consiste em considerar que todos os grupos de palavras têm pelo menos uma ocorrência, incluindo aqueles que não ocorrem nenhuma vez. Este método designa-se por alisamento adicionar um.

Como exemplo, e considerando para o caso dos *bigramas*, em primeiro lugar é realizada uma matriz de ocorrência em que se considera que todos os pares de palavras vão ocorrer pelo menos uma vez. Depois actualizam-se os pares de ocorrências com os valores de ocorrência que existem no corpus. Seguidamente adicionam-se 1 a todos os valores de ocorrência da matriz, incluindo os nulos. Depois normalizam-se os valores, actualizando o número total de ocorrências com os valores acrescentados.

Este método, no entanto, não é muito utilizado pois não apresenta bons resultados em termos de utilização e pode provocar uma grande diferença em relação às frequências originais quando se realiza o alisamento. O principal problema reside no facto de adicionar o valor 1 às contagens. Se fosse adicionado um valor mais pequeno, o problema era atenuado.

O segundo método de alisamento designa-se por desconto Witten-Bell [Witten, 1991]. Este método, embora mais complexo que o anterior, baseia-se no conceito de modelar a primeira ocorrência dos *Ngramas* para estimar a ocorrência dos que ainda não ocorreram.

Assim a probabilidade de ocorrência de um *ngrama* que ainda não ocorreu é modelada com a ajuda dos *Ngramas* que apenas ocorreram uma vez. Se se considerar o exemplo dos *bigramas*, o valor de estimação de ocorrência dos *bigramas* que não têm nenhuma ocorrência é dado pela contagem dos *bigramas* que ocorreram apenas uma vez. O valor final é normalizado com o número de *bigramas* observados e dividido por todos os possíveis *bigramas* que não têm ocorrência.

Este método faz com que o cálculo da probabilidade de ocorrência seja dependente do histórico de ocorrência de palavras. Também as palavras que ocorrem em menos combinações de *bigramas* tendem a ter menos *bigramas* não vistos que as que entram em mais combinações de *bigramas* diferentes.

O terceiro método de alisamento designa-se por desconto Good-Turing [Good, 1953]. Este método, embora mais complexo que o anterior, tem como ideia principal a de voltar a calcular as probabilidades de ocorrência para atribuir valores aos *Ngramas* que são nulos e aos que têm

valores muito baixos com base no número de *Ngramas* com elevados valores de probabilidade de ocorrência.

A ideia é calcular as frequências de ocorrência dos valores de ocorrência c, ou seja, calcula-se para c=1 quantos tipos de *Ngramas* apenas ocorrem uma vez. Para c=i calcula-se quantos *Ngramas* diferentes têm esse valor de ocorrência i. Constrói-se assim uma tabela de frequências de ocorrência em que para c=0 tem-se a contagem de *Ngramas* com frequência de ocorrência nula. Como se espera, quanto menor é o c maior é a frequência de ocorrência.

Se se considerar para o exemplo de *bigramas*, a contagem revista dos *bigramas* que nunca ocorreram é calculado dividindo o número de *bigramas* que ocorreram uma vez pelo número total de *bigramas* que nunca ocorreram.

Na prática, este desconto não é aplicado para todos os valores de c. As frequências maiores, onde c>k (em que k representa o valor a partir do qual não são recalculados) são assumidas como fiáveis. Katz [Katz, 1987] sugere que k tome o valor 5.

Os métodos de desconto vistos, permitem lidar com os *Ngramas* que não têm nenhuma ocorrência no modelo. Existem no entanto outros processos que permitem lidar com o facto de não existir um *ngrama* específico e calcular a sua frequência de ocorrência com base nos *Ngramas* de ordem *n-1*. Se se considerar o caso dos *trigramas*, o cálculo é efectuado com base nos *bigramas*. Para o caso de existir um *bigrama* que não tenha ocorrências, então a sua frequência de ocorrência é baseada na frequência de ocorrências das palavras simples. Existem duas formas de aplicar este processo por interpolação apagada ou backoff.

O quarto método de alisamento, designado por backoff, é um método não linear introduzido por Katz em 1987 [Katz, 1987]. Para salientar a diferença deste processo, e se se considerar o exemplo em que se tem *trigramas* com contagem diferente de 0, então apenas se tem em consideração a frequência de ocorrência dos *trigramas*. Se se pretender calcular a ocorrência de um trigrama que não tem nenhuma ocorrência consideram-se os *bigramas*. Se se pretender calcular um *bigrama* que não tenha ocorrência consideram-se as ocorrências das palavras simples.

2.4 A Ferramenta CMUSLM

No âmbito deste trabalho, o modelo de língua utilizado foi gerado utilizando a ferramenta de domínio público *Carnegie Mellon University Statistical Language Modeling* [Clarkson & Rosenfeld, 1997] que abreviadamente se designa por *CMUSLM* ou *CMU*.

Esta ferramenta consiste num conjunto de programas para facilitar a construção e teste dos modelos de língua de *Ngramas* [Clarkson & Rosenfeld, 1997]. A Figura 1 mostra, através de um diagrama de actividade descrito na linguagem UML, a forma de utilização desta ferramenta. Este diagrama corresponde a uma simplificação e adaptação da figura original de [Clarkson & Rosenfeld, 1997].

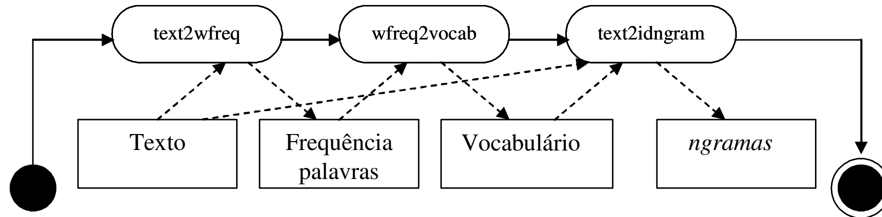


Figura 1 – Uso da ferramenta *CMUSLM*

Para criar os modelos estatísticos de língua são necessários três passos. No primeiro passo é gerada, a partir do corpus de texto, a estatística de ocorrência das palavras. Este passo tem como resultado um ficheiro com todas as diferentes palavras contidas no corpus de texto e a respectiva frequência de ocorrência.

No segundo passo é gerado, a partir do ficheiro obtido no primeiro passo, o vocabulário do corpus de texto. O vocabulário corresponde às palavras do corpus de texto ordenadas por ordem alfabética. Neste segundo passo é ainda possível definir o limite máximo de palavras, excluindo assim as palavras menos frequentes.

No terceiro passo são construídos os modelos de *Ngramas*. Os ficheiros de entrada necessários para gerar os modelos são o corpus de texto inicial e o vocabulário obtido no segundo passo. Os modelos de *Ngramas* gerados são compostos por três ficheiros, em que cada um representa um grupo de *n* palavras e a respectiva probabilidade de ocorrência. Para *n* igual a um corresponde à frequência de ocorrência das palavras, para *n* igual a dois corresponde à frequência de ocorrência dos pares de palavras e para *n* igual a três corresponde à frequência de ocorrência dos trios de palavras.

2.5 Modelos Utilizados

As primeiras experiências foram realizadas com um modelo de língua composto por frequências de palavras simples e frequências de pares de palavras. Este modelo de língua é composto por 65.817 palavras e respectivas frequências e por 80.847 pares de palavras e

respectivas frequências. Este modelo de língua faz parte da aplicação *Eugénio* [Garcia & Oliveira, 2001] que realiza a predição de palavras e foi concebido para acelerar o processo de escrita a pessoas com limitações motoras.

Depois de várias experiências verificou-se que a maioria dos poemas continha palavras que não estavam contempladas no modelo de língua. Assim foram realizadas experiências com um modelo com maior número de palavras e com frequências de ocorrência para grupos de três e quatro palavras.

Este modelo de língua, com a probabilidade de ocorrência das palavras, foi obtido a partir de uma interpolação de dois modelos de língua, um deles obtido com base em jornais recolhidos da Web e o outro obtido com base nos textos das transcrições disponíveis dos programas noticiosos de televisão.

Os valores de probabilidade de ocorrência das palavras, permitem ordenar as palavras por ordem decrescente de probabilidade. A Tabela 1 mostra o número de ocorrências de cada um dos grupos de palavras do modelo.

Grupos	Número
1 Palavra	57.564
2 Palavras	5.724.469
3 Palavras	11.095.964
4 Palavras	6.731.820

Tabela 1 – Contagem dos grupos de palavras do modelo de língua.

No modelo apresentado, são usadas 57.564 palavras diferentes. Se se contabilizarem todos os grupos de palavras obtém-se um total de 23.609.817 ocorrências. Para manipular este número de palavras e uniformizar o modo de acesso aos modelos foi utilizado um motor de base de dados para armazenar esta informação.

Com este modelo, o número de palavras que não existiam no modelo diminui bastante mas, mesmo assim existiam palavras que não estavam no modelo. Para estes casos, a frequência de ocorrência destas palavras é calculada com o método backoff.

2.6 Exemplos do Modelo Utilizado

O modelo de língua utilizado está organizado da seguinte forma:

- Palavras simples com valor de frequência e factor de backoff;

- Grupos de duas palavras com valor de frequência e factor de backoff;
- Grupos de três palavras com valor de frequência e factor de backoff;
- Grupos de quatro palavras com valor de frequência.

Para cada um dos casos foi tirada uma amostra de 5 linhas da tabela correspondente e cada uma das tabelas mostra um grupo de palavras, simples, de duas palavras, de três palavras e de quatro palavras.

Em todos os casos, o número que aparece em primeiro lugar corresponde à frequência de ocorrência da linha correspondente no formato de logaritmo na base 10.

Nos casos das palavras simples, grupos de dois e de três, também aparece o factor de backoff que serve para calcular a frequência quando um *Ngrama* não existe.

Os grupos de quatro palavras não apresentam valores de backoff porque não foram usados modelos com grupos de 5 palavras.

Frequência	Palavra	Factor backoff
...	...	...
-6,48570900000	abadessa	-0,10814660000
-5,94152900000	abadia	-0,29602020000
-6,48570900000	abadias	-0,14308460000
-6,13287000000	abafada	-0,34335430000
-5,91964100000	abafado	-0,31932020000
...	...	...

Figura 2 – Palavras Simples com Valor de Frequência e Factor de backoff

Frequência	Palavra1	Palavra2	Factor backoff
...	...	...	...
-6,71511600000	a	abade	0,00000000000
-5,88075100000	a	abadessa	-0,01240277000
-5,38470200000	a	abadia	-0,12448850000
-5,47629800000	a	abafar	-0,04290840000
-5,23826200000	a	abaixo	0,00000000000
...	...	...	...

Figura 3 – Grupos de Duas Palavras com Valor de Frequência e Factor de backoff

Frequência	Palavra1	Palavra2	Palavra3	Factor backoff
...	...	...	...	...
-2,66555000000	a	a	cidade	-0,08316695000
-2,13003300000	a	a	cinco	-0,07574962000
-3,25000700000	a	a	cinquenta	-0,35222700000
-3,39421900000	a	a	classificação	0,00000000000
-3,37385500000	a	a	colocar	0,00000000000
...	...	...	...	...

Figura 4 – Grupos de Três Palavras com Valor de Frequência e Factor de Backoff

Frequência	Palavra1	Palavra2	Palavra3	Palavra4
...	...	...	...	...
-2,57598900000	a	a	bola	enquanto
-2,48734000000	a	a	bola	fernando
-2,20488200000	a	a	bola	Foi
-2,02271700000	a	a	bola	Fonte
-2,50851900000	a	a	bola	Já
...	...	...	...	...

Figura 5 – Grupos de quatro palavras com valor de frequência

Se, por exemplo, se pretender calcular a frequência de ocorrência da palavra abadessa, pode-se fazê-lo, bastando para isso calcular 10 ^ -6,485709 = 3,268e-7.

Esta forma de cálculo aplica-se a todos os grupos de palavras existentes e a frequência resultante está compreendida entre 0 e 1.

Para os casos em que não existe o grupo de palavras para o qual se pretende saber a frequência usou-se o processo de cálculo com base em backoff.

Seguidamente apresenta-se o exemplo para um grupo de três palavras:

Considera-se wd1, wd2 e wd3, respectivamente a 1ª palava, a 2ª palavra e a 3ª palavra.

Considera-se p_3 (wd1, wd2, wd3) a probabilidade do trio w1, w2 e w3.

Considera-se bo_wt_2(w1, w2) o factor de backoff do par w1, w2.

Então:

```
p(wd3|wd1,wd2)= Se(existe trigrama)
                  p_3(wd1,wd2,wd3)
                Senão
                  Se(existe par w1,w2)
                    bo_wt_2(w1,w2)+p(wd3|wd2)
```

```
                  Senão
                    p(wd3|wd2)
p(wd2|wd1)= Se(existe par)
              p_2(wd1,wd2)
            Senão
              bo_wt_1(wd1)+p_1(wd2)
```

Todas as probabilidades e factores de backoff estão no formato $\log_{10}$.

3 CONCEITOS DA POÉTICA PORTUGUESA

3.1 Introdução

Neste capítulo são apresentadas as definições encontradas na pesquisa bibliográfica efectuada de modo a definir a implementação dos conceitos no sistema, para realizar quer a classificação quer a sugestão das palavras finais.

Por vezes foram encontradas várias definições para o mesmo conceito pelo que se optou por apresentar apenas aquelas definições que contribuíram para a implementação desses conceitos no sistema.

3.2 Acentuação das Palavras

A acentuação das palavras permite determinar o tipo das palavras que são usadas nos poemas e é necessária na classificação dos poemas. Para se classificar a palavra quanto à sua acentuação são necessários dois passos:

- em primeiro lugar, decompor a palavra em sílabas;
- em segundo lugar, verificar a posição da sílaba tónica (ou acentuada) da palavra.

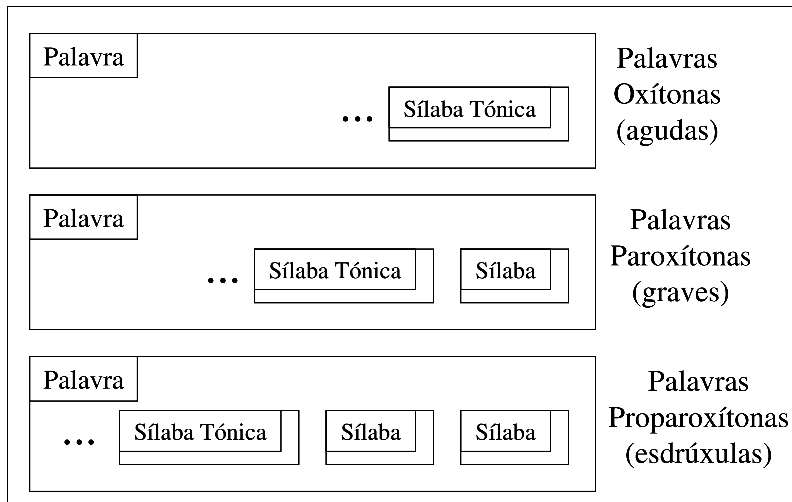


Figura 6 – Classificação das palavras.

A sílaba acentuada pode encontrar-se em três posições (Figura 6):

- Última sílaba;
- Penúltima sílaba;
- Antepenúltima sílaba.

No caso da sílaba acentuada se encontrar na última sílaba a palavra é aguda ou oxítona. No caso de se encontrar na penúltima sílaba a palavra é grave ou paroxítona. No caso de se encontrar na antepenúltima sílaba a palavra é esdrúxula ou proparoxítona.

Como exemplo, as palavras 'acentuação', 'vistas' e 'última' têm, respectivamente, os tipos aguda, grave e esdrúxula.

3.3 Transcrição Fonética das Palavras

Foi usado o alfabeto fonético para o dialecto padrão do português europeu [SAM-PA], que é apresentado na Tabela 2. Para cada som do alfabeto fonético existe um exemplo de palavra e respectiva transcrição fonética.

Um alfabeto fonético é usado para representar os sons das palavras e possibilitar a comparação de equivalência de sons entre palavras. O alfabeto fonético permite, dada uma palavra, representar a respectiva transcrição fonética fazendo corresponder à palavra o seu som. A partir das transcrições fonéticas das palavras é possível determinar se duas palavras têm o mesmo som, possibilitando a detecção da rima.

Som / Exemplo		Som / Exemplo		Som / Exemplo		Som / Exemplo		Som / Exemplo	
i	idade [**i**dad@]	**i~**	inchou [**i~**So]	**p**	pago [**p**agu]	**f**	fora [**f**Or6]	**l**	leal [**l**jal~]
e	erros [**e**RuS]	**e~**	enjoa [**e~**Zo6]	**b**	burra [**b**uR6]	**v**	vala [**v**al6]	**l~**	algés [a**l~**ZES]
E	erva [**E**rv6]	**6~**	anda[**6~**d6]	**t**	ti [**t**i]	**s**	seco [**s**eku]	**L**	bolha [bo**L**6]
a	acre [**a**kr@]	**o~**	bombo [b**o~**bu]	**d**	dar [**d**ar]	**z**	zeros [**z**EruS]		
6	abelha [**6**b**6**L**6**]	**u~**	um [**u~**]	**k**	campo [**k**6~pu]	**S**	chapa [**S**ap6]	**r**	par [pa**r**]
@	bate [bat@]	**j~**	dêem [de6~**j~**]	**g**	galho [**g**aLu]	**Z**	gelar [**Z**@lar]	**R**	rato [**R**atu]
O	orca [**O**rk6]	**w~**	adão [6d6~**w~**]						
o	ouro [**o**ru]							**m**	mudo [**m**udu]
u	uno [**u**n**u**]							**n**	nobre [**n**Obr@]
j	ideais [id**j**a**j**S]							**J**	cunho [ku**J**u]
w	aos [a**w**S]								

Tabela 2 – Alfabeto fonético para o dialecto padrão do português europeu SAM-PA.

Os sons são condicionados pelo aparelho fonador, sendo distinguidos consoante o papel das cavidades nasais, o modo de articulação e o papel das cordas vocais. A Tabela 2 está organizada em cinco colunas. Na primeira coluna encontram-se as vogais orais e na segunda coluna as vogais nasais. Na terceira coluna encontram-se as consoantes orais oclusivas, na quarta coluna encontram-se as consoantes orais constritivas frincativas. Na quinta coluna encontram-se três grupos, distinguindo-se em primeiro lugar as orais constritivas laterais, em segundo lugar as orais constritivas vibrantes e em terceiro as consoantes nasais sonoras.

Para cada som é mostrado um exemplo de palavra com a respectiva transcrição fonética. Destacando dois exemplos da tabela, pode-se verificar que o som que é representado pela letra 'i' existe na palavra 'idade' cuja respectiva transcrição fonética 'idad@' inclui este som no início da palavra. O som que é representado pelas letras 'o~' existe na palavra 'bombo' cuja respectiva transcrição fonética 'bo~bu' inclui este som.

3.4 Rima

Como já foi referido anteriormente, a verificação da rima tem em conta a transcrição fonética das palavras. Para o realizar, as palavras são convertidas no conjunto de caracteres do alfabeto fonético que correspondem aos sons da palavra.

Quando se fala em transcrição fonética, existem dois aspectos que são necessários ter em conta:

- Existem palavras com múltiplas transcrições fonéticas;
- É necessário tratar as palavras que não estão contidas no léxico.

Um dos critérios usados, para decidir qual a transcrição fonética a escolher, tem como base a categoria gramatical da palavra depois de efectuar a análise sintáctica da frase que contém a palavra. No âmbito deste trabalho optou-se por escolher a transcrição fonética mais comum e assim utilizar apenas uma.

Quando uma palavra não existe no léxico, a transcrição fonética dessa palavra é fornecida pela aplicação externa e é adicionada ao léxico do sistema. Quando a palavra já existe no léxico então é usada a informação aí existente. Desta forma consegue-se optimizar o acesso às palavras que já existem no léxico, pois antes de ser armazenada é feito um pré processamento à palavra de modo a optimizar o acesso. A aplicação externa gera as transcrições fonéticas com base num conjunto de regras.

No português europeu a ortografia pode ser considerada de base essencialmente fonológica, ou seja, existe uma elevada regularidade entre a ortografia e a fonética [Oliveira, 1996]. Esta também é uma das razões pela qual os dicionários de rima apenas baseados nas letras terminais das palavras mesmo assim conseguem obter bons resultados de consulta.

A transcrição fonética só por si não é suficiente para realizar a verificação da rima. Para se conseguir implementar a rima, também é necessário incluir na transcrição fonética a indicação do som que corresponde à vogal tónica da palavra.

O resultado da transcrição fonética é obtido a partir da aplicação externa *Leia* [Oliveira, 1996] e o resultado obtido pode ser parametrizável na invocação. Teve de se activar o parâmetro que assinala a vogal acentuada da palavra.

A aplicação *Leia* é baseada no sistema *DIXI*, que foi o primeiro sistema de síntese de fala a partir de texto desenvolvido de raiz para a língua portuguesa [Oliveira, 1996]. A transcrição fonética é realizada quer por consulta de um dicionário, quer por um conjunto de regras. No contexto deste trabalho, não se pretende gerar fala, mas a utilização da transcrição fonética para verificação de rima é o que permite identificar as palavras consoantes.

A indicação da vogal acentuada é assinalada pelo símbolo " e todos os caracteres que aparecem após este caracter correspondem à transcrição fonética da terminação da palavra que é comparada para efeitos de rima.

Palavra	Transcrição Fonética
Eu	"ew
Não	n"6~w~
Tenho	t"6Ju
Vistas	v"iSt6S
Largas	l"arg6S
Sabedoria	s6b@dur"i6

Tabela 3 – Exemplo de transcrições fonéticas.

A Tabela 3 apresenta um conjunto de exemplos de transcrições fonéticas de palavras extraídas de uma quadra. Destacando um dos exemplos apresentados pode-se verificar que a transcrição fonética da palavra 'Não' corresponde a (n"6~w~).

Se se analisarem as terminações fonéticas das palavras 'Vistas' e 'Largas', para as comparar em termos de rima, obtêm-se respectivamente as terminações fonéticas 'iSt6S' e 'arg6S'. Como se pode verificar elas são diferentes, o que significa que as palavras não rimam entre si.

3.5 Divisão em Sílabas Gramaticais

A divisão silábica das palavras implementada na aplicação *Leia* tem como base um conjunto de 11 regras que contemplam como fronteira de sílaba uma sequência consoante vogal, tendo o cuidado de não separar os grupos indivisíveis (pr, gl, etc.) e os dígrafos (nh, ch, etc.) [Oliveira, 1996]. Para realizar a divisão silábica das palavras, apenas foi necessário usar a mesma aplicação externa *Leia*, alterando os parâmetros de invocação da aplicação.

Na divisão silábica, é usado o símbolo '$' como separador de sílabas e é também assinalada a vogal acentuada com o mesmo símbolo anteriormente definido. O processamento das vogais acentuadas que é utilizado na aplicação externa Leia corresponde a um formato interno composto pelas letras minúsculas entre a e z e pelas marcas de acento agudo (´), circunflexo (^) e til (~) imediatamente a seguir à respectiva vogal e pelo acento grave (`) antes da vogal. A cedilha é representada pelo símbolo (,) depois da letra C.

Palavra	Divisão Silábica
Eu	eu
Não	n"a~o
Tenho	t"e$nho
Vistas	v"is$tas
Largas	l"ar$gas
Sabedoria	sabedo$r"i$a

Tabela 4 – Exemplo de divisões silábicas.

A Tabela 4 exemplifica alguns resultados de divisão silábica. Destacando dois exemplos apresentados pode-se verificar que a divisão silábica da palavra 'Não' corresponde a (n"a~o). Pode-se ainda verificar que a palavra 'Não' é composta por uma única sílaba e corresponde a uma palavra aguda, por ser acentuada na última sílaba. Já a palavra 'Sabedoria' é composta por cinco sílabas gramaticais, é acentuada na penúltima sílaba, o que corresponde a uma palavra grave.

Na divisão silábica é a indicação da vogal acentuada que permite verificar se se trata de uma palavra aguda, grave ou esdrúxula, bastando para isso contar o número de sílabas que aparecem após a indicação de sílaba tónica. No caso da palavra 'Sabedoria' podemos contar mais uma sílaba após a indicação de sílaba tónica.

No léxico, estas palavras são representadas com os caracteres com os respectivo acentos, ou seja, é feita a conversão das letras mais os acentos que se obtêm em letras com acentos. Para depois ficar de acordo com o léxico é necessário a conversão de (n"a~o) para (n"ão).

3.6 Divisão em Sílabas Métricas

As sílabas métricas são as sílabas contadas nos versos tal como são apercebidas pelo ouvido. Para realizar a divisão dos versos em sílabas métricas é necessário ter em conta o ritmo do verso que define o seu período rítmico.

A principal diferença em termos de contagem é que as sílabas métricas apenas contabilizam o número de sílabas até à última sílaba acentuada da palavra. Existem ainda outros processos que alteram o número de sílabas e que fazem aumentar ou diminuir o número de sílabas métricas.

Existem duas regras que resumem as várias situações:

- Quando uma palavra termina em vogal e a primeira sílaba da palavra seguinte também começa por vogal, desde que não sejam ambas tónicas, dá-se uma junção das duas numa só;
- Os hiatos podem transformar-se em ditongos e os ditongos podem transformar-se em hiatos.

Como exemplo, apresenta-se um verso de Fernando Pessoa dividido em sílabas métricas:

- "Qual/quer/ coi/sa em/ mi/nha al/(ma)".

O verso é composto por seis sílabas métricas. As junções das sílabas 'sa em' e 'nha al' são um exemplo da primeira regra. No primeiro caso, em que as vogais são diferentes designa-se por *elisão* e no segundo caso em que as vogais são iguais designa-se por *crase*.

No caso do exemplo 'coisa em' e dada a fusão entre o 'a' e o 'e' obtém-se a seguinte divisão 'coi/sem' mas dependendo da pronúncia também se pode obter o ditongo 'ái' e neste caso fica a divisão 'coi/sáim', designando-se neste caso por *sinalefa* em que a fusão entre o 'a' e o 'e' forma o ditongo 'ái'.

Outro exemplo que ilustra a segunda regra, são os versos em que entra a palavra 'glórias'. Gramaticalmente a palavra tem três sílabas 'gló/ri/as' mas se se juntar o hiato 'i' e 'a' em que as vogais pronunciadas separadamente passam a pronunciar-se como uma vogal mais uma semi vogal tem-se o ditongo 'ia' e passa a palavra a contabilizar apenas duas sílabas 'gló/rias'. Este caso designa-se por *sinérese*. O inverso também acontece com menos frequência e dá-se o nome de *diérese*.

Pode ainda dar-se a supressão de sons no início, meio ou fim da palavra designados respectivamente por *aférese*, *síncope* e *apócope*. As palavras 'estamos', 'coroa' e 'mármore' são exemplo respectivamente '/stamos', 'c/roa' e 'mármor/'.

Uma hipótese, para implementar estas regras, é a de adicionar ao texto da transcrição fonética as marcas com as várias hipóteses de divisão silábica das palavras [Mamede et al., 2004].

No trabalho realizado contabilizam-se as sílabas métricas tendo em conta a contagem de sílabas até à sílaba acentuada da última palavra do verso.

3.7 Categoria Morfológica das Palavras

Um dos processos que permite seleccionar e excluir, logo à partida, grande parte das palavras são as categorias gramaticais das palavras. O objectivo é a partir de uma frase incompleta verificar quais as categorias possíveis para a próxima palavra e assim restringir o conjunto de palavras possíveis.

Com base no conjunto de categorias obtido, são seleccionadas apenas as palavras que pertencem a esse conjunto de categorias. O objectivo desta selecção é reduzir o número de palavras.

A análise sintáctica é realizada em várias fases e com diferentes aplicações externas. Os dados de entrada são uma frase incompleta e os dados de saída são o conjunto de classes possíveis para a próxima palavra. A Figura 7 resume o processo descrito.

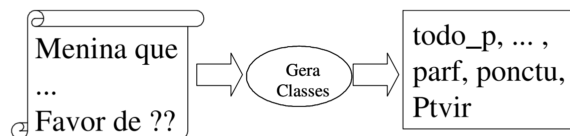


Figura 7 – Módulo externo de geração de classes.

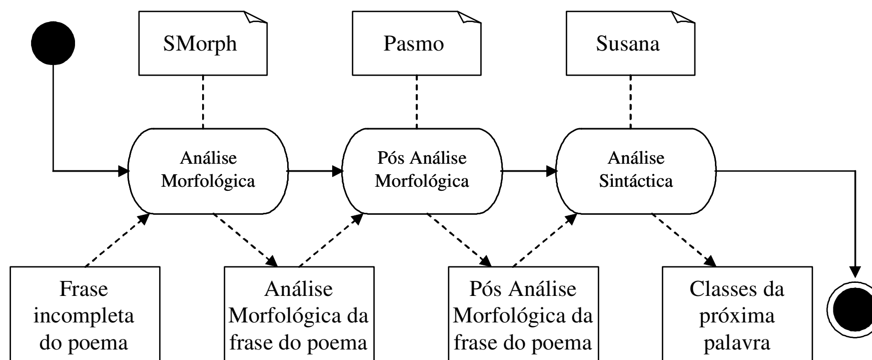


Figura 8 – Funcionamento interno do módulo externo de geração de classes.

A Figura 8 ilustra o funcionamento interno do módulo externo de geração de classes através de um diagrama de actividade descrito na linguagem UML.

No primeiro passo é utilizada a aplicação *Smorph* [Ait-Mokhtar, 1998]. Este passo tem como entrada a frase incompleta do poema e obtém como resultado a análise morfológica das palavras que compõem a frase e que inclui, a título de exemplo, a categoria, o número e o género. Este resultado vai ser passado para a próxima fase. Este primeiro passo usa um dicionário.

O segundo passo consiste em realizar a pós análise morfológica. A aplicação *Pasmo* [Paulo & Mamede, 2001], responsável por este passo, filtra alguns dos lemas possíveis e altera o formato de algumas palavras. Este passo reescreve o texto de acordo com as regras e também agrupa as palavras em frases.

O terceiro passo corresponde a uma análise sintáctica da frase que permite determinar para essa frase quais as classes possíveis para a próxima palavra [Hagège, 2000] e [Batista, 2002]. Para tal utilizou-se a aplicação Susana que agrupa os constituintes da frase e gera como resultado as várias classes possíveis para a próxima palavra.

3.8 Conceitos Estruturais

As definições dos conceitos da poética portuguesa que servem de base a esta proposta são um pouco vagas e, por vezes, recorrem a sentimentos e imagens para caracterizar os conceitos. Algumas definições são definidas à custa de outras e por vezes existem definições circulares dos conceitos.

A preocupação principal para compreender e implementar estes conceitos foi a de procurar várias definições para o mesmo conceito e conseguir extrair a parte que permite a sua automatização, não esquecendo de registar os respectivos compromissos adoptados. As diferentes definições são confrontadas e comparadas.

O primeiro conceito a definir é o de poema. São várias as definições possíveis, que vão desde as mais subjectivas às mais objectivas. Começando por uma das mais subjectivas, das analisadas, pode-se dizer que *poema* é "*precisamente uma técnica linguística de produção dum tipo de consciência que o espectáculo do mundo não produz ordinariamente*", citação de *Jean Cohen*, *Structure du langage poétique*, 1966, [Moisés, 1974].

Outra definição com o mesmo nível de subjectividade é a seguinte: "*Considera-se poema toda composição literária de índole poética*" [Moisés, 1974].

Das definições analisadas, a mais objectiva foi a de que o poema é "*um organismo verbal que contém, suscita ou segrega poesia*", citação de Octavio Paz, El Arco y la Lira, 1956, [Moisés, 1974] e que define poema com base na definição de poesia.

Também foram várias as definições de poesia que foram analisadas. Foram escolhidas duas definições de poesia sendo a primeira mais subjectiva e a segunda mais objectiva.

Como primeira definição de poesia tem-se: "*é a comunicação, estabelecida com meras palavras, de um conhecimento de índole muito especial: o conhecimento de um conteúdo, psíquico*

tal como é: ou seja, de um conteúdo psíquico como um todo particular, como síntese intuitiva, única, do conceptual-sensorial-afetivo", citação de Teoria de la Expressión Poética, 4ª ed., 1969, [Moisés, 1974].

A segunda definição de poesia é: "*Se eu chamar prosa a um discurso mínimo, veículo mais económico do pensamento, e chamar, a, b, c, a atributos particulares da linguagem, inúteis mas decorativos, tais como o metro, a rima ou o ritual das imagens, toda a superfície das palavras se encaixará na dupla equação de M. Jourdain:*

Poesia = Prosa + a + b + c

Prosa = Poesia - a - b - c

Daí resulta evidentemente que a Poesia é sempre diferente da Prosa. Mas tal diferença não é de essência, é de quantidade", citação de O Grau Zero da Escritura, tr. br., 1971, [Moisés, 1974].

A partir desta última definição conclui-se que poesia é diferente de prosa e a diferença encontra-se nos atributos particulares da linguagem, como são o metro e a rima.

Foi ainda analisada outra definição que colide com a segunda definição de poesia apresentada que afirma que "*existem poemas sem poesia, e a poesia pode surgir no âmbito de um romance ou de um conto*" [Moisés, 1974]. Esta definição baseia-se mais na primeira definição de poesia apresentada, ou seja, na comunicação estabelecida entre o poeta e o leitor, no conteúdo psíquico, conceptual, sensorial e afectivo. No âmbito desta dissertação não se considera a poesia moderna onde a fronteira entre estes conceitos é ainda mais difusa.

Não tendo a ambição de levar estas definições ao extremo, nem o desejo de contemplar todas as hipóteses possíveis de poemas, e olhando para estas definições de poesia de um ponto de vista prático e funcional, surge a necessidade de restringir o domínio da definição, bem como dos poemas possíveis de serem utilizados. O primeiro compromisso que se adopta, é o de admitir que o domínio dos poemas que se pretende alcançar é aquele em que existe poesia. Os textos poéticos têm uma estrutura de escrita bem definida e regras de construção bem definidas que obedecem às normas impostas pela tradição e respeitam os aspectos formais da escrita, como são exemplo o metro e a rima.

Das definições de poema e poesia apresentadas, conclui-se que, do ponto de vista estrutural, um poema está organizado em estrofes. Assim, para as definições ficarem completas é necessário definir estrofe.

Uma estrofe é "*um conjunto de, versos, solidários pelo ritmo e inseparáveis pelo pensamento*", citação de Amorim de Carvalho, Tratado de Versificação Portuguesa, 1941, [Moisés, 1974]. Embora um pouco filosófica, esta definição é complementada pela seguinte definição: "*Por estrofe entende-se cada uma das secções que constituem um poema, ou seja cada*

agrupamento de versos, rimados ou não, com unidade de conteúdo e de ritmo" [Moisés, 1974]. Conclui-se que as estrofes correspondem aos grupos de linhas que constituem o poema.

Como as estrofes se organizam em versos, torna-se necessário definir verso. Se se tiver em conta a utilização mais antiga desta palavra, "*Inicialmente significava em latim, a volta que dava a charrua ao fim de cada sulco, mas depois passou a aplicar-se, por extensão, ao próprio sulco; finalmente, por metáfora, ganhou o sentido de linha de escrita, que finalmente se especializou no de linha de escrita em poesia, composta por um número determinado de sílabas*" [Coelho, 1987].

Outra definição mais resumida aponta que um "*verso é a sucessão de sílabas ou fonemas formando unidade rítmica e melódica, correspondente a uma linha do poema. Cada verso subdivide-se ainda em subunidades caracterizadas pelo agrupamento de sílabas chamado de pé na versificação greco-latina*" [Moisés, 1974]. Assim se obtém uma definição que para estar completa necessita apenas de definir o significado das subunidades que compõem o verso.

A estrutura do poema, apresentada na Figura 9, resume as definições que foram apresentadas.

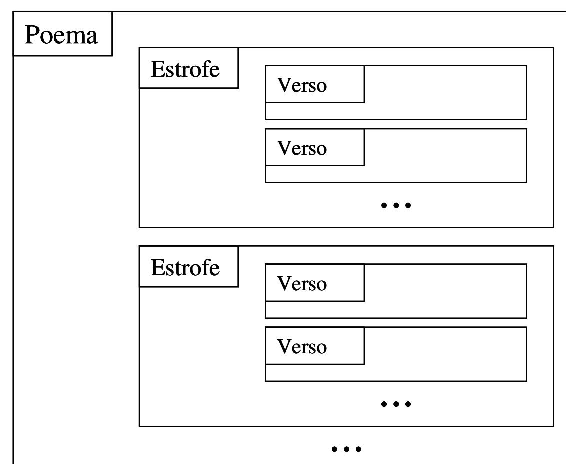


Figura 9 – Estrutura do poema.

Um poema corresponde a um texto, em que as mudanças de linha definem o fim de cada verso e uma linha vazia define as mudanças de estrofe. Uma estrofe corresponde, portanto, a um conjunto de linhas que termina numa linha em branco. As definições adoptadas permitem definir a estrutura de um poema e assim automatizar a sua classificação.

Um verso pode ser constituído por uma palavra só ou por várias. Os elementos do verso que foram considerados para implementar a sua classificação foram o número de sílabas e a rima.

Segundo a definição de verso apresentada anteriormente ficaram por definir as subunidades que compõem o verso. Os versos podem ser subdivididos usando diferentes métricas: em sílabas

ou em pés. Um pé "*designa a unidade rítmica e melódica do verso, composta de um grupo de sílabas. Remonta aos gregos e romanos, que mediam os versos em sequências temporais separadas por intervalos regulares. Cada sequência, ou célula métrica, compunha-se de duas ou mais sílabas, cuja mensuração se fazia pelo tempo despendido na sua prolação*" [Moisés, 1974].

Conclui-se, portanto, que os pés correspondem a agrupamentos de sílabas que têm em conta o tempo despendido na prolação. Os versos superiores a 5 sílabas podem ser subdivididos em hemistíquios, que correspondem a metade do verso, e são assinalados por uma pausa ou cesura.

As pausas ou cesuras correspondem a cortes no interior do verso e a diferença entre pausa e cesura é que a cesura é fixa enquanto a pausa pode variar com a pessoa.

Nesta proposta, apenas se adoptou como subunidade do verso a sílaba, uma vez que corresponde à unidade mais elementar de decomposição da palavra, podendo a partir dela obterem-se as outras subdivisões. Existem, como foi visto anteriormente, dois tipos de sílabas que podem ser tidos em conta quando se fala numa composição poética: as sílabas gramaticais e as sílabas métricas. As sílabas gramaticais dividem as palavras segundo as leis da gramática e as sílabas métricas têm em conta a forma como são apreciadas pelo ouvido. Por estarem sujeitas a contracções e serem contabilizadas até à sílaba tónica da última palavra (sistema que foi iniciado entre nós, por Feliciano de Castilho[Areal, 2000]), o número de sílabas métricas pode ser igual ao número de sílabas gramaticais, mas tipicamente é inferior. Nesta dissertação foi usada a contagem de ambas as sílabas gramaticais e métricas sem contemplar as junções das vogais.

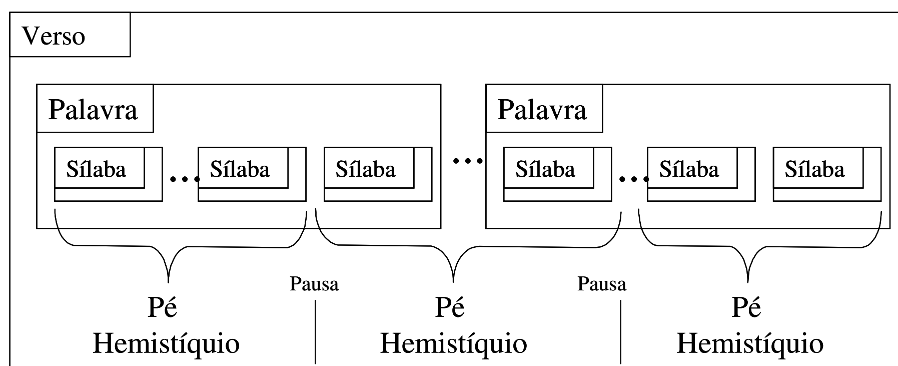


Figura 10 – Estrutura do verso.

A estrutura do verso apresentada na Figura 10 esquematiza a definição apresentada. Em resumo, cada verso está decomposto em palavras que, por sua vez, se dividem em sílabas. As sílabas que podem ser consideradas são as sílabas gramaticais ou, obter a partir destas, as sílabas métricas e os pés. Tanto as sílabas métricas como os pés têm em conta o tempo de prolação. A decomposição de um verso em sílabas métricas designa-se por escansão.

Como exemplo de decomposição em sílabas, apresenta-se um verso de Fagundes Varela:

- "*Que prende o céu à terra e a terra aos anjos*", [Areal, 2000]

E respectivas decomposições silábicas:

- Sílabas gramaticais: "Que/ pren/de/ o/ céu/ à/ ter/ra/ e/ a/ ter/ra/ aos/ an/jos". (15 sílabas)
- Sílabas métricas: "Que/ pren/de o/ céu/ à/ ter/ra e a/ ter/ra aos/ an/(jos)". (10 sílabas)

O próximo conceito a definir, com vista à sua automatização, é a rima. "*Depara-se-nos uma rima (final) quando, em duas ou mais palavras, a última vogal acentuada, com tudo o que se lhe segue, tem idêntica sonoridade*", citação de Análise e Interpretação da Obra Literária, 1958, vol. I, [Moisés, 1974]. Por outras palavras, rima é a correspondência sonora entre as sílabas finais dos versos.

Quando a correspondência de sons a partir da vogal tónica é perfeita, incluindo vogais e consoantes, a rima diz-se consoante. Se pelo contrário, a correspondência de sons se verifica apenas entre as vogais, a rima é toante. As palavras 'faz' e 'papás' ou 'inclina' e 'pequenina' são um exemplo de rima consoante. As palavras 'faz' e 'lá' ou 'inclina' e 'filha' são um exemplo de rima toante.

A rima pode ainda designar-se por rima *rica* no caso de se dar entre palavras de classe gramatical diferente ou rima *pobre* quando se dá entre palavras de classe gramatical igual. Servem de exemplo de rima rica as palavras 'sepulcrais' (adjectivo) e 'mais' (advérbio) ou 'desordem' (substantivo) e 'mordem' (verbo). Servem de exemplo de rima pobre as palavras 'vês' (verbo) e 'lês' (verbo) ou 'miserável' (adjectivo) e 'inseparável' (adjectivo).

A Figura 11 ilustra a definição de rima adoptada.

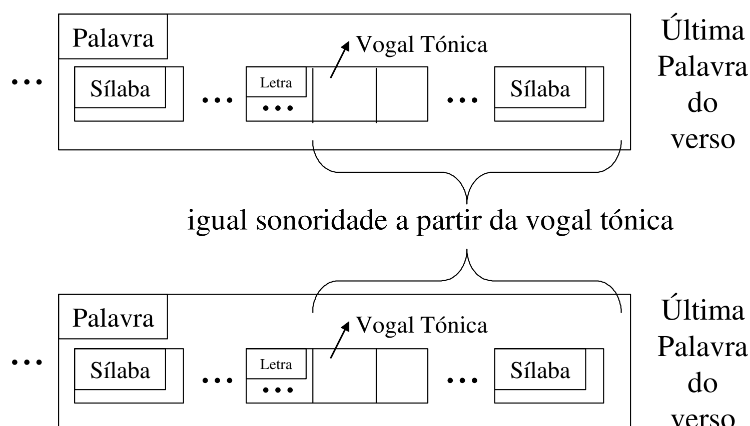


Figura 11 – Esquematização de rima.

O processo responsável por automatizar a detecção da rima tem em conta apenas as últimas palavras de cada verso e, para estas palavras, sinaliza a última vogal acentuada. Pode-se concluir, em resumo, que duas palavras rimam quando, a partir da vogal tónica (inclusive) e com tudo o que se lhe segue, as palavras têm igual sonoridade.

Também surgiu a necessidade de limitar o âmbito da definição de rima e em termos de implementação consideram-se apenas rima consoante, ou seja, aquela em que existe uma correspondência total a partir da sílaba tónica.

Todas as definições apresentadas serviram de base para realizar a aquisição dos poemas, na medida em que permitem implementar um algoritmo que realiza a aquisição dos poemas e identifica os conceitos estruturais e de rima dos poemas a classificar.

3.9 Regras de Classificação de Poemas

Quando se constrói um poema, é necessário ter em conta alguns factores que determinam diferentes tipos de poema: existem poemas em que o número de versos que constituem as estrofes é constante; existem poemas em que o número de sílabas de cada verso é constante; e existem poemas em que os versos que rimam entre si poderão apresentar-se com diferentes configurações. A Tabela 5 resume as diferentes classificações quanto ao número de versos.

Nº de Versos	Designações possíveis	Designação Adoptada
1	monótico	monótico
2	dístico, parelha ou pareado	dístico
3	trístico ou terceto	terceto
4	tetrástico, quadra ou quarteto	quadra
5	pentástico, quinteto ou quintilha	quintilha
6	hexástico, sextilha, sexteto ou septena	sextilha
7	heptástico, sétima, septilha, septena ou hepteto	sétima
8	octástico ou oitava	oitava
9	nona, eneagésima ou novena	nona
10	decástico, década ou décima	décima
n	n versos	n versos

Tabela 5 – Classificação das estrofes quanto ao nº versos.

As estrofes, também designadas por estâncias, correspondem a um agrupamento de versos e consoante a distribuição dos versos pelas estrofes assim se obtêm diferentes tipos de estrofe com diferentes designações. Os números de versos mais comuns são de 2, 3, 4, 5, 6, 8 e 10 versos, como se pode ver na Tabela 5.

Para cada valor de número de versos, em cada uma das linhas da Tabela 5, existem diferentes designações possíveis e a sua utilização varia de autor para autor. Foi escolhida a designação mais comum entre as várias hipóteses para os valores entre 1 e 10. Para as estrofes com mais de 10 versos adoptou-se a regra geral de colocar o número de versos seguido da palavra verso(s). Por exemplo, uma estrofe com 13 versos aparece a designação '13 versos'.

Outro factor que permite distinguir diferentes tipos de poemas é o número de sílabas que compõem o verso. Existem na língua portuguesa versos desde uma a treze sílabas, sendo os mais usados os de 5, 6, 7, 10 e 12. A Tabela 6 resume as várias classificações possíveis.

Nº de Sílabas	Designações possíveis	Designação Adoptada
1	1 sílaba	1 sílaba
2	dissílabo ou bissílabo	bissílabo
3	trissílabo, quebrado de redondilha maior, redondilho quebrado ou cola	trissílabo
4	tetrassílabos	tetrassílabos
5	pentassílabo ou redondilha menor	pentassílabo
6	hexassílabo, heróico quebrado ou heróico menor	hexassílabo
7	heptassílabo ou redondilha maior	heptassílabo
8	octossílabo	octossílabo
9	eneassílabo, verso de gregório de matos	eneassílabo
10	decassílabo, heróico, sáfico ou provençal	decassílabo
11	hendecassílabo ou verso de arte maior	hendecassílabo
12	alexandrino	alexandrino
20	vintissílabos	vintissílabos
n	n sílabas	n sílabas

Tabela 6 – Classificação dos versos quanto ao nº sílabas.

Também na Tabela 6 existem alguns valores de números de sílabas que apresentam várias designações possíveis e a sua utilização varia de autor para autor. Adoptou-se, também neste caso,

a designação mais usual. Para os versos que não têm designação na tabela, adoptou-se a regra geral de colocar o número de sílabas seguido da palavra sílaba(s).

Os versos de uma e duas sílabas são raros. No entanto apresentam-se dois poemas, a título de curiosidade, de Casimiro de Abreu [Areal, 2000], um com versos de uma sílaba e o outro com versos de duas sílabas:

Poema de uma Sílaba	Poema de duas Sílabas
Amo	Na valsa
Gemo	Tão falsa,
Clamo	Corrias,
Tremo	Fugias,
	Ardente,
	Contente,
	Serena,
	Sem Pena
	De mim

Ambos os exemplos apresentam uma contabilização das sílabas métricas dos versos. Como se pode verificar, os versos de uma sílaba iniciam com a sílaba acentuada, não existindo nenhuma outra antes desta. Nos versos de duas sílabas apenas existe uma sílaba antes da sílaba tónica.

Para classificar os versos quanto ao tipo de rima distinguem-se dois tipos: os versos rimados; os versos soltos ou brancos (sem rima). Cada um dos casos caracteriza diferentes tipos de poemas. Isto significa que a rima, embora não seja obrigatória existir num poema, é um factor de classificação.

O tipo da última palavra permite classificar o tipo de rima, pois, como foi apresentado anteriormente, varia com a posição da sílaba acentuada da última palavra. Os versos podem ser:

- Versos agudos - quando as palavras utilizadas na rima são agudas;
- Versos graves - quando as palavras utilizadas na rima são graves;
- Versos esdrúxulos - quando as palavras utilizadas na rima são esdrúxulas.

As rimas com palavras esdrúxulas são valorizadas, pois apresentam um maior grau de dificuldade.

Também a disposição estrófica da rima em relação aos versos que enlaça determina a classificação da rima. Para representar a disposição estrófica da rima são utilizadas letras do alfabeto que representam a rima da última palavra do poema. Para cada verso com rima diferente das já existentes é atribuída uma letra começando na letra A. Depois, em todos os versos com igual rima é usada a mesma letra.

Quando, por exemplo, se diz que uma estrofe é do tipo (ABAB), isto significa que a estrofe é composta por quatro versos e como a primeira letra é igual à terceira sabe-se que o primeiro verso rima com o terceiro. Aplicando a mesma regra, pode-se concluir que o segundo verso rima com o quarto e obviamente tem rima diferente do primeiro e terceiro.

A Tabela 7 resume as várias hipóteses possíveis de classificação quanto à forma como se encadeiam os versos que rimam.

Designação	Descrição
emparelhadas	Quando os versos que rimam se encontram juntos e aos pares (ABB CDD EFF)
cruzadas	Quando entre dois versos que rimam se encontra outro de diferente rima (ABCB ou ABAB)
abraçadas	Quando entre dois versos que rimam se encontram dois versos de diferente rima (ABBA ou ABCA)
interpolada	Quando entre dois versos que rimam se encontram três ou mais versos de diferente rima (ABBBA ou ABCDA)
seguida	Quando rimam mais de dois versos seguidos (AAA)
monórrimos	Quando está sujeito a uma só rima que pode também ser cruzada (ABCB)

Tabela 7 – Classificação quanto à posição relativa da rima e dos versos que enlaça.

Como exemplo destaca-se a rima cruzada da forma ABAB muito usada nas quadras populares e em que o primeiro verso rima com o terceiro e o segundo rima com o quarto.

3.10 Tipos de Poemas

A partir das definições que foram adoptadas, é possível realizar a classificação para diferentes tipos de poemas. Entende-se por diferentes tipos de poemas, aqueles que apresentam diferente número de versos que compõem as estrofes, ou poemas que apresentem diferente número de sílabas que formam os versos, ou poemas que apresentem rima com diferentes categorias de palavras finais ou, ainda, poemas com diferentes configurações na forma como a rima é enlaçada.

Para além destas possíveis diferenças, foram ainda utilizados poemas de teste com diferentes origens com o objectivo de diversificar, quer no vocabulário quer na estrutura. Os poemas realizados por crianças em idade escolar, por exemplo, não apresentam uma estrutura bem definida e homogénea. Pelo contrário, o número de versos por estrofe varia durante o poema, tal como o número de sílabas em cada verso. Abaixo apresentam-se dois exemplos de estrofes realizadas por crianças de 9 anos [Jorge et al., 2000]:

"*menina que leva a vida*
sentadinha a escrever,
faça favor de ensinar,
eu também quero aprender.

Brincar, brincar
és para brincar
e alegrar"

A primeira estrofe é composta por 4 versos e a segunda por três versos. O número de sílabas em cada verso varia. Na segunda estrofe, por exemplo, o primeiro verso tem 4 sílabas gramaticais ou métricas e o segundo tem 5 sílabas gramaticais ou métricas, e o terceiro verso tem 4 sílabas gramaticais ou métricas. Nestes exemplos o facto da última sílaba ser a sílaba tónica faz com que o número de sílabas gramaticais e métricas sejam iguais se não se considerar as junções das vogais.

Em relação à rima, na primeira estrofe foram utilizadas palavras graves e agudas e a forma como a rima foi enlaçada foi o 2º verso a rimar com o 4º verso. Já na segunda estrofe foram utilizadas palavras agudas e a forma como se enlaçou a rima foi a de todos os versos rimarem entre si. Também é comum nos poemas de crianças a tendência para colocar todos os versos com a mesma rima (rima seguida).

As duas primeiras estrofes da obra Os Lusíadas, de Camões, mostram outro exemplo de poema utilizado no teste do sistema:

“*As armas e os barões assinalados*
Que da ocidental praia lusitana
Por mares nunca de antes navegados
Passaram ainda além da Taprobana,
Em perigos e guerras esforçados
Mais do que prometia a força humana,
E entre gente remota edificaram
Novo Reino, que tanto sublimaram;

E também as memórias gloriosas
Daqueles Reis que foram dilatando
A Fé, o Império, e as terras viciosas
De África e de Ásia andaram devastando,
E aqueles que por obras valerosas
Se vão da lei da morte libertando,
Cantando espalharei por toda parte,
Se a tanto me ajudar o engenho e arte.”

Neste exemplo, ambas as estrofes têm 8 versos (oitavas). Também o número de sílabas em cada verso é constante contando-se sempre 12 sílabas gramaticais ou 10 sílabas métricas (decassílabo ou heróico). Neste caso estas estrofes também tomam a designação de *oitava-rima*. A forma como os versos são enlaçados é sempre a mesma, sendo os 6 primeiros versos de rima cruzada e os dois últimos de rima emparelhada, ou seja, (A B A B A B C C).

As quadras populares, também correspondem a um tipo de poema em que é constante o número de versos, o número de sílabas e o enlace da rima. Foram seleccionadas 3 quadras de António Aleixo para o ilustrar:

“*Eu não tenho vistas largas,*
Nem grande sabedoria,
Mas dão-me as horas amargas
Lições de Filosofia.

Há luta por mil doutrinas.
Se querem que o mundo ande,
Façam das mil pequeninas
Uma só doutrina grande.

Quando os Homens se convençam
Que à força nada se faz,
Serão felizes os que pensam
Num mundo de amor e paz."

Neste exemplo as estrofes são compostas de 4 versos (quadra), os versos são compostos por 8 sílabas gramaticais ou 7 sílabas métricas (redondilha maior), quanto à rima as palavras utilizadas variam entre agudas e graves e o enlace tem sempre a mesma configuração de rima cruzada, ou seja, (A B A B).

4 O SISTEMA LUCAS

4.1 Introdução

Para validar as definições anteriormente descritas, e com o objectivo de construir um sistema que cumpra os objectivos iniciais, foi construído um demonstrador a que se deu o nome de *LuCas*. O nome do sistema foi inspirado no nome do poeta Luís de Camões. O sistema *LuCas* realiza, por um lado, a classificação de poemas que são fornecidos pelo utilizador, e por outro lado, sugere as palavras finais dos versos, quando o utilizador está a construir poesia, tendo em conta uma estrutura pré-definida de poema escolhida pelo utilizador.

Foram feitas algumas opções de implementação com vista a facilitar, por um lado, o funcionamento em vários sistemas operativos e, por outro, possibilitar a visualização num browser. Para tentar satisfazer da melhor forma possível estas duas opções optou-se pela linguagem de programação Java, que corre em diferentes sistemas operativos. A aplicação foi construída sob a forma de uma Applet, possibilitando assim o seu funcionamento num browser.

Em termos de arquitectura, também foram feitas algumas opções de desenho para permitir alguma flexibilidade e adaptabilidade às alterações de requisitos funcionais. O sistema implementado está organizado em módulos funcionais concebidos para serem o mais independentes possível, com vista à reutilização das suas funcionalidades por outras aplicações.

No contexto do grupo de investigação, as aplicações externas são disponibilizadas através da interface baseada em browser GalInHa [Matos et Al., 2003] inspirada plataforma Galaxy [Seneff et al., 1998], que corresponde a uma arquitectura aberta para construção de sistemas de diálogo. Esta infra-estrutura distribuída tem uma aproximação em que os módulos são independentes entre si e podem ser acrescentados ou removidos da infra-estrutura, permitindo, no caso em que são adicionados à infra-estrutura disponibilizar o acesso às suas funcionalidades estando assim prontos a funcionar.

O Galaxy foi introduzido pela primeira vez em 1994 e consistia numa plataforma de testes para tecnologias de língua falada. Foi remodelada em 1998 com o intuito de ser uma plataforma de referência que usa uma linguagem de *script* para controlo de fluxo [Seneff et al., 1998]. O controlo dos diversos servidores que estão disponíveis no Galaxy é feito através de um *hub* que controla o fluxo de informação entre eles [Seneff et al., 1999]. Esta plataforma providencia, por um lado, uma boa ferramenta para desenvolver sistemas e, por outro, a de configurar e avaliar esses sistemas [Polifroni & Seneff, 2000].

Houve neste trabalho a preocupação de divisão das funcionalidades por vários módulos para possibilitar a futura integração no GalInHa [Matos et Al., 2003]. Uma das vantagens da divisão em módulos é permitir melhorar o funcionamento de um determinado módulo sem ter de estar preocupado com os efeitos colaterais nos outros módulos e desde que se mantenha a interface do módulo. A outra vantagem é que se podem substituir os módulos independentemente uns dos outros.

Houve ainda a preocupação de que cada módulo tivesse uma interface muito bem definida para permitir disponibilizar de forma fácil as funções por ele implementadas. Os processos de coordenação do sistema *LuCas* realizam as funções principais do sistema. Para o realizar activam através da interface dos módulos as funções necessárias para satisfazer os pedidos do utilizador.

4.2 Arquitectura do Sistema

O sistema *LuCas* é composto por sete módulos e dois processos de coordenação. Dois dos módulos realizam a interface entre o sistema e as várias aplicações externas, dois outros realizam as operações relacionadas com o léxico e a sugestão de palavras e os restantes três módulos realizam as operações de interface com o utilizador, identificação dos conceitos e identificação das regras da poética portuguesa. Relativamente aos processos de coordenação, um é responsável pela classificação dos poemas [Araújo & Mamede, 2002] e outro pela sugestão das palavras finais dos versos.

A Figura 12 ilustra os módulos e os processos que compõem o sistema e as respectivas interligações entre eles.

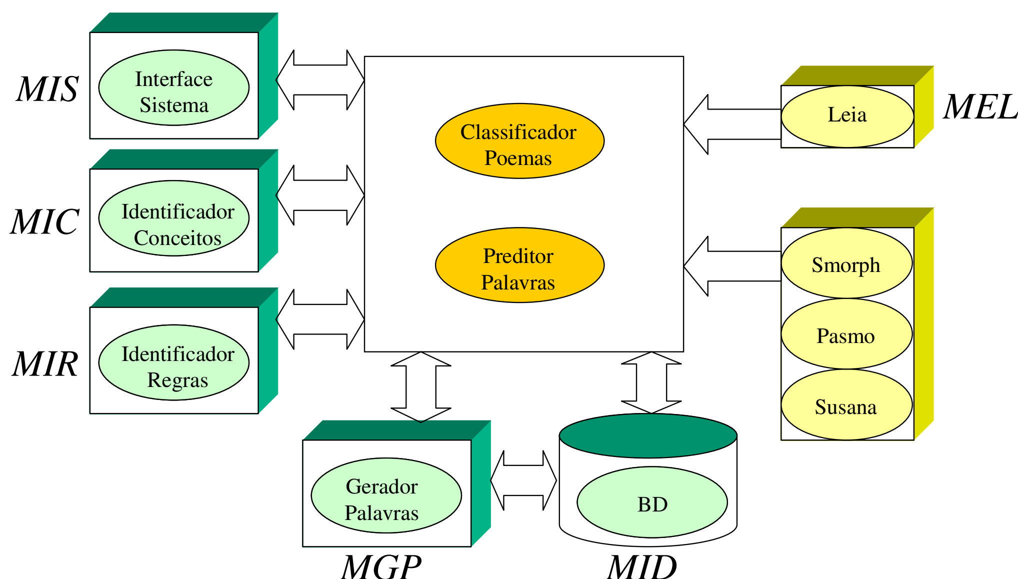


Figura 12 – Arquitectura do Sistema *LuCas*.

Seguidamente apresenta-se um resumo de cada módulo bem como as suas respectivas funções de interface.

Começando pelos três módulos que se apresentam na Figura 12 do lado esquerdo, o *Módulo de Interface do Sistema* (MIS) é responsável pelas operações de interface do sistema e o *Módulo Identificador de Conceitos* (MIC) e o *Módulo Identificador de Regras* (MIR) são responsáveis pela identificação dos conceitos e regras da poética portuguesa existentes nos poemas. Estes módulos partilham uma estrutura de dados comum, interna do sistema, onde o poema é armazenado e a respectiva informação adicional resultante do pré-processamento do poema.

Quando é realizada a classificação dos poemas a informação de classificação é adicionada na estrutura de dados interna [Araújo & Mamede, 2002]. Também a sugestão de palavras usa a informação da estrutura de dados interna.

A opção de partilhar uma estrutura de dados pelos três módulos, adoptando-se uma manipulação do tipo quadro preto, melhora o desempenho do sistema, pois evita ter de passar várias vezes a mesma informação entre os vários módulos. Por outro lado, esta utilização restringe a generalidade e independência entre módulos, pois aumenta o acoplamento entre eles.

O MIS realiza as funções de interface com o utilizador. Estas operações incluem editar poemas, guardar e ler em ficheiros os poemas editados e receber os comandos efectuados pelo utilizador. Os comandos aceites pelo sistema permitem validar as palavras do poema, classificar o poema e sugerir palavras finais dos versos. Foram incluídas no sistema algumas funções associadas à rima das palavras, que permitem a pesquisa de palavras segundo a sua rima.

Este módulo também é responsável pela visualização dos resultados de classificação, visualização das palavras finais dos versos sugeridas pelo sistema e das palavras que rimam com uma determinada palavra. É responsável ainda pela configuração do sistema e pelas mensagens de ajuda que guiam o utilizador nos comandos possíveis de ser efectuados.

Este módulo disponibiliza, na sua interface, as seguintes funções:

validaPoema(Poema)-	esta função devolve um valor binário de verdadeiro ou falso, indicando se o poema passado como argumento na forma de texto é válido ou não;
processaPoema(Poema)-	esta função realiza o pré-processamento do poema passado como argumento na forma de texto, decompondo-o em linhas e estrofes e armazenando-o numa estrutura interna que servirá de base aos módulos seguintes;
mostraClassificação()-	esta função converte o resultado da classificação do poema que está armazenado na estrutura interna do sistema num formato texto de fácil percepção e apresenta ao utilizador;
mostraSugestão()-	esta função apresenta ao utilizador as palavras que foram sugeridas pelo sistema em relação ao poema que estava a ser construído. O número de palavras visualizadas pode ser configurado no sistema;
mostraRima()-	esta função apresenta ao utilizador as palavras que rimam com uma determinada palavra;
mostraRimas()-	esta função apresenta ao utilizador, para um conjunto de palavras, um grupo com todas as que rimam com a primeira e outro grupo com todas as que não rimam.

O MIC é responsável por assinalar para um poema fornecido os conceitos da poética portuguesa. O processamento é realizado sobre a estrutura de dados interna que contém o poema pré-processado e o resultado é o registo dos conceitos estruturais e de rima anteriormente descritos que são adicionados na mesma estrutura interna. A informação de identificação de conceitos serve de base para depois serem aplicadas as regras de classificação.

Este módulo disponibiliza na sua interface a seguinte função:

identificaConceitos()-	esta função adiciona ao poema pré-processado a informação dos conceitos da poética portuguesa.

O MIR é responsável pela implementação das regras de classificação dos poemas e acrescenta na estrutura que contém o poema a classificar, a respectiva informação de classificação, com base nas regras de classificação anteriormente descritas, e na informação de identificação de conceitos adicionada pelo módulo anterior. Após ser acrescentada a informação de classificação, é convertida para depois ser mostrada ao utilizador pelo módulo de interface do sistema.

Este módulo disponibiliza na sua interface a seguinte função:

classificaPoema()- esta função realiza a classificação do poema com base na implementação das regras descritas e adiciona a informação de classificação ao poema.

Seguidamente descrevem-se os módulos: *Módulo de Interface de Dados* MIL e o *Módulo Gerador de Palavras* (MGP) apresentados na Figura 12 e correspondem respectivamente ao módulo que implementa o léxico e ao módulo responsável pela geração de palavras. Estes módulos fazem a interface entre o sistema e o repositório de dados do sistema. O léxico inclui todas as palavras que são aceites pelo sistema e o módulo gerador de palavras optimiza os acessos ao módulo do léxico.

O *Módulo de Interface de Dados* (MID) realiza a interface entre o sistema e a base de dados do sistema. A base de dados é composta pelo léxico e pelos modelos de língua. Este módulo baseia-se na utilização de um sistema de gestão de base de dados relacional, que armazena as palavras que são aceites pelo sistema e armazena os modelos estatísticos de língua usados na sugestão de palavras.

Uma das vantagens de utilização do léxico na classificação de poemas é a de permitir realizar a detecção de erros ortográficos dos poemas, quer daqueles que se pretende classificar, quer dos que se encontram em construção. Outra vantagem de utilização do léxico é poder utilizar vários léxicos alternativos consoante o contexto que se pretende analisar e consoante a utilização que se pretende fazer. Por exemplo, se se pretender classificar poemas mais antigos, então tem de se acrescentar no léxico as palavras antigas. Se se pretender utilizar como ferramenta de construção de poemas nas escolas para crianças, então poderá ser usado um léxico adaptado para esse contexto.

Este módulo disponibiliza na sua interface as seguintes funções:

existePalavra(Palavra)- esta função devolve um valor binário de verdadeiro ou falso caso exista ou não no léxico a palavra passada por argumento;

existePalavras(Texto)- esta função devolve o conjunto de palavras do texto passado por argumento que não existem no léxico;

inserePalavra(Palavra)- esta função acrescenta a palavra passada por argumento no léxico;

removePalavra(Palavra)- remove a palavra passada por argumento do léxico.

O MGP realiza as operações responsáveis pela sugestão de palavras. A interacção entre este módulo e o léxico é directa, por forma a optimizar o desempenho do sistema. Para realizar a sugestão de palavras existe a hipótese de configurar o sistema de forma a seleccionar o modo que melhor se adapta à sugestão pretendida. A sugestão de palavras tem em conta a parte já escrita do poema e a configuração escolhida. O seu funcionamento está baseado numa função que toma em conta as várias hipóteses de selecção das palavras.

Este módulo disponibiliza na sua interface a seguinte função:

próximaPalavra(n)- esta função sugere *n* palavras para completar um verso. O valor *n* é passado por argumento à função.

Por último, os módulos *Módulo de Interface Externa Leia* (MEL) e o *Módulo de Interface Externa Gerador de Classes* MEC, representados na Figura 12 a amarelo, realizam a interface com as aplicações externas.

O MEL realiza a interface entre o sistema e a aplicação externa *Leia*. Este módulo obtém a transcrição fonética e a divisão silábica das palavras e disponibiliza na sua interface as seguintes funções:

transcriçãoFonética(Palavra)- esta função devolve a transcrição fonética de uma palavra passada por argumento (inclui a indicação de sílaba tónica);

divisãoSilábica(Palavra)- esta função devolve a divisão silábica da palavra passada por argumento (inclui a indicação de sílaba tónica).

O MEC realiza a interface entre o sistema e as aplicações externas *Smorph, Pasmo e Susana*. Este módulo gerador de classes obtém, para uma frase incompleta (sem a última palavra), um conjunto de classes possíveis para essa palavra e disponibiliza na sua interface a seguinte função:

classesPossíveis(Frase)- esta função devolve um conjunto de classes possíveis para a próxima palavra que pode formar a frase passada por argumento.

A base de dados do sistema permite que o sistema funcione de forma autónoma sem aceder às aplicações externas. Foram adicionadas funções que permitem armazenar a informação que é gerada pelas aplicações externas na base de dados.

4.3 Processos de Coordenação

O *processo classificador de poemas* (PCP) realiza a classificação de poemas. Os módulos utilizados para a classificação de poemas são o MIS, MIC, MIR e MIL.

O PCP realiza a classificação de poemas em 4 etapas.

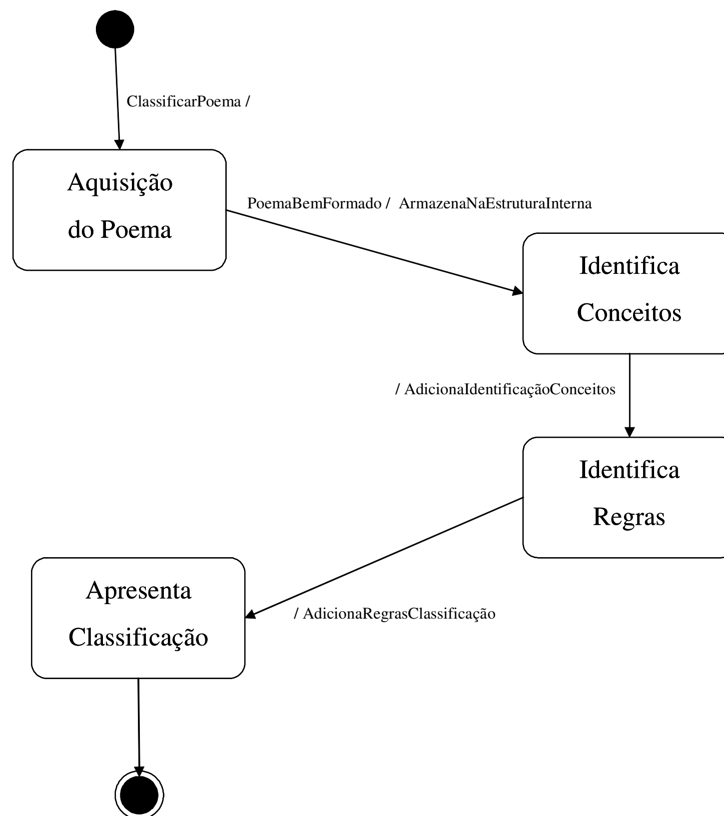


Figura 13 – Diagrama de estados do processo de classificação.

A Figura 13 resume, na linguagem UML, os estados do processo de classificação.

Na primeira etapa, o MIS é responsável por realizar a aquisição do poema para uma estrutura interna e armazenar o poema num formato que identifica as linhas do poema, destacando as palavras finais dos versos. Este formato simplifica a etapa de classificação seguinte. Pode,

opcionalmente, ser realizada a verificação de vocabulário do poema, utilizando as funções do *MIL*, para garantir que as palavras que compõem o poema existem no léxico.

Na segunda etapa o MIC acrescenta a informação de identificação de conceitos. Esta informação inclui a identificação das estrofes, dos versos e das rimas. Também é acrescentada a informação da transcrição fonética e divisão silábica das palavras.

Na terceira etapa o MIR tem como base a informação de identificação de conceitos da etapa anterior e realiza a classificação do poema com base em regras que incluem a classificação das estrofes, a classificação dos versos e a classificação da rima.

Na quarta e última etapa, o MIS é responsável por apresentar o resultado final de classificação ao utilizador.

Seguidamente é apresentado, na Figura 14, um diagrama de actividade na linguagem UML que corresponde ao algoritmo simplificado que realiza a aquisição de poemas e armazena os conceitos estruturais do poema relativos ao verso e estrofe para permitir a posterior classificação.

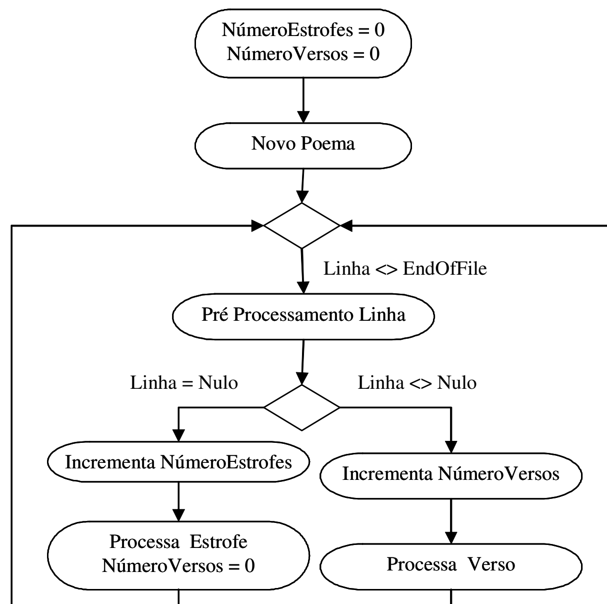


Figura 14 – Diagrama de actividade de identificação de versos e estrofes.

Este algoritmo está implementado no MIC. Começa por iniciar as variáveis `NúmeroEstrofes` e `NúmeroVersos` com 0 para depois realizar um ciclo de leitura das linhas do poema. Por cada linha lida diferente de nulo incrementa a variável `NúmeroVersos` que indica o número de versos numa estrofe. Por cada mudança de linha em branco no texto é incrementada a variável `NúmeroEstrofes` que indica o número de estrofes total do poema e é guardado o

número de versos da estrofe. Por cada mudança de estrofe é iniciada a variável `NúmeroVersos` com 0.

O *processo preditor de palavras* (PPP) realiza a predição das palavras finais dos versos. Este processo utiliza todos os módulos do sistema para realizar a predição das palavras finais dos versos. Podem ocorrer diferentes situações de poemas que originam diferentes critérios de predição. O primeiro factor que condiciona a predição das palavras está relacionado com o facto de se estar a considerar ou não uma estrutura definida de poema.

No caso em que não existe uma estrutura definida de poema, o critério de selecção baseia-se nos modelos estatísticos de língua que permitem sugerir as palavras, ordenadas por ordem decrescente de frequência de ocorrência das palavras.

Quando o utilizador não impõe uma estrutura de poema pré-definida existem vários factores que são tidos em conta para sugerir as palavras. O primeiro critério corresponde ao número de sílabas dos versos que apenas selecciona as palavras com um número específico de sílabas. O segundo critério corresponde à disposição estrófica da rima, que apenas selecciona as palavras com uma determinada rima.

Em ambos os casos, foram realizadas experiências utilizando as categorias gramaticais das palavras. A partir de uma frase incompleta são sugeridos os tipos de palavras possíveis para a próxima palavra. As experiências realizadas não foram conclusivas, pois o número de classes sugeridas restringe pouco o número total de palavras não chegando a filtrar 50% das palavras.

A Figura 15 mostra, através de um diagrama de actividade descrito na linguagem UML, o processo de escolha do método de selecção das palavras a sugerir. Existem três formas de selecção de palavras: a primeira por número de sílabas; a segunda por rima e a terceira por frequência de ocorrência. A escolha depende da configuração que o utilizador escolheu e depende da situação encontrada no poema.

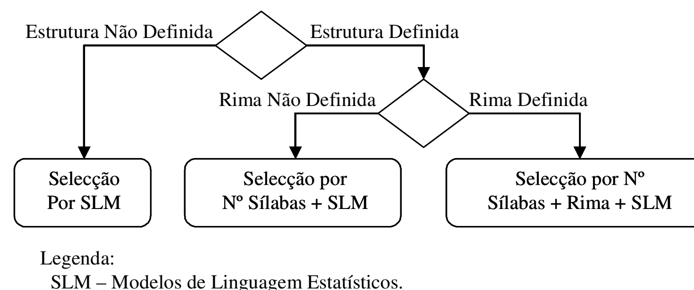


Figura 15 – Diagrama de actividade de predição de palavras.

4.4 Arquitectura da Base de Dados

Inicialmente, a base de dados era composta pelo léxico. Este léxico permite validar o vocabulário utilizado na realização dos poemas e assim ter no sistema a funcionalidade de corrector ortográfico. Mais tarde foi também utilizado para guardar a informação que se obtém a partir das aplicações externas e os modelos de língua.

Foi desenvolvido um conjunto de funcionalidades para permitir adaptar e complementar a informação relativa às transcrições fonéticas, às divisões silábicas e às categorias das palavras que são geradas pelas aplicações externas e acrescentar essa informação na base de dados. Foi ainda criada a estrutura de dados necessária para suportar os modelos de língua utilizados pelo sistema na sugestão de palavras.

O diagrama de classes apresentado na Figura 16, descrito na linguagem UML, mostra a estrutura de dados do léxico.

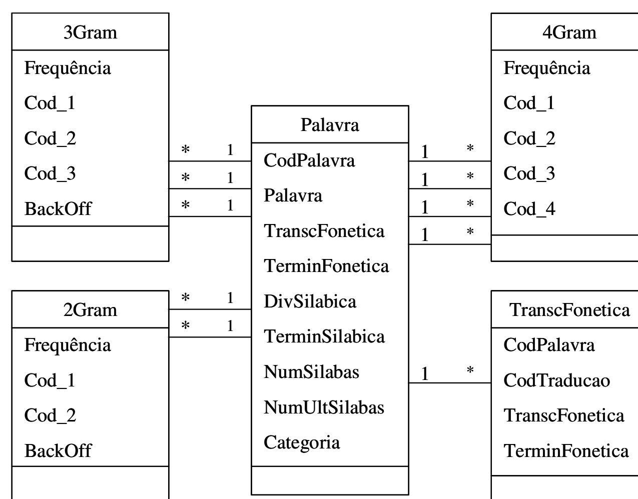


Figura 16 – Modelo de dados do léxico.

Este modelo, como já foi referido anteriormente, foi implementado com base num sistema de base de dados relacional para optimizar o acesso aos dados.

Começando pela classe principal, foi criada uma entidade que se designou por `Palavra` e que serve para armazenar as palavras que são válidas no sistema. Por cada palavra que é inserida no léxico é atribuído um código único `CodPalavra` que a identifica univocamente e que é incrementado por cada nova inserção de palavra. A palavra propriamente dita é guardada no atributo `Palavra`. Por consequência, todas as palavras que não existirem no léxico ou são fornecidas pelas aplicações externas ou são adicionadas manualmente.

As primeiras experiências que foram realizadas utilizaram um léxico de 10000 palavras com as respectivas transcrições fonéticas e divisões silábicas. Após algumas experiências de classificação realizadas com poemas de crianças e quadras populares de António Aleixo, verificou-se que existiam muitas palavras que não estavam contempladas no léxico. Foram então acrescentadas as palavras que ainda não existiam no léxico e que eram necessárias para poder classificar os poemas. Após esta operação, o léxico ficou com 11505 palavras.

Para o caso da transcrição fonética, existe a necessidade de armazenar várias transcrições fonéticas por cada palavra. Tendo em conta as 11505 palavras do léxico, o número de transcrições fonéticas por cada palavra varia entre 1 e 7. A Tabela 8 mostra a distribuição das palavras por número de transcrições fonéticas.

Descrição	Número
N.º Total de palavras	11505
1 Transcrição Fonética	10131
2 Transcrições Fonéticas	1214
3 Transcrições Fonéticas	81
4 Transcrições Fonéticas	75
5 Transcrições Fonéticas	2
6 Transcrições Fonéticas	1
7 Transcrições Fonéticas	1

Tabela 8 – Distribuição do número de transcrições fonéticas.

Foi acrescentada, no modelo de dados apresentado na Figura 16, a classe `TranscFonetica` que é responsável pelo seu armazenamento. O atributo `TranscFonetica` permite guardar a transcrição fonética gerada pela aplicação externa *Leia* e o atributo `TerminFonetica` foi adicionado para conter a transcrição fonética correspondente apenas à parte final das palavras com o objectivo de optimizar o desempenho das pesquisas por rima.

Outro factor que contribuiu para melhorar o desempenho do sistema foi adicionar os atributos `TranscFonetica` e `TerminFonetica` na classe `Palavra` que correspondem,

respectivamente, à transcrição fonética mais comum da palavra e à respectiva transcrição fonética a partir da vogal acentuada. Esta optimização justifica-se uma vez que cada palavra tem pelo menos uma transcrição fonética. Esta optimização evita que o sistema aceda a duas entidades passando a aceder a apenas a uma.

Esta optimização não invalida as duas utilizações, ou seja, existe a possibilidade de configurar se se pretende fazer a pesquisa das palavras que rimam usando apenas a transcrição fonética mais comum, e desta forma usar apenas a entidade Palavra, ou usar todas as possíveis transcrições fonéticas de cada palavra, acedendo inevitavelmente às duas entidades.

Para se ter uma ideia do número de palavras existente com igual terminação fonética foi realizada a contagem das palavras agrupadas por igual terminação fonética.

Para um total 11505 palavras que correspondem ao dicionário acrescentado, existem 1825 terminações fonéticas diferentes. A Tabela 9 mostra as primeiras 10 terminações fonéticas com maior número de palavras.

Terminação da Palavra	Transcrição Fonética	Número Palavras
ão	6~w~	425
ar	ar	397
ado	adu	353
or	or	265
ada	ad6	263
ia	i6	194
ou	o	185
ados	aduS	184
ente	e~t@	176
ores	or@S	152
...	...	...

Tabela 9 – Contagem das palavras com igual terminação fonética.

Como se pode concluir, a sugestão de 425 palavras, no caso mais desfavorável, é um número demasiado elevado para ser útil.

No caso da divisão silábica, e como apenas existe uma possível divisão por cada palavra, acrescentaram-se os atributos `DivSilabica`, `TerminSilabica`, `NumSilabas` e `NumUltSilabas` que permitem armazenar, respectivamente, a divisão silábica gerada pela aplicação externa *Leia*, a divisão silábica a partir da vogal acentuada, o número total de sílabas

gramaticais da palavra e o número de sílabas a partir da vogal acentuada. Estes atributos foram acrescentados por questões de eficiência na pesquisa de palavras, pois podem ser obtidos a partir da divisão silábica da palavra. Os atributos `NumUltSilabas` e `TerminSilabica` permitem determinar se a palavra é aguda, grave ou esdrúxula.

Tendo em conta as 11505 palavras, foram agrupadas as palavras com o mesmo número de sílabas e o resultado é mostrado na Tabela 10.

Número Sílabas	Número Palavras
1	277
2	2543
3	4232
4	3052
5	1108
6	245
7	41
8	6
9	1

Tabela 10 – Contagem das palavras com igual número de sílabas.

As palavras em maior número são as de 3 sílabas correspondendo, a 37% do total das palavras. Seguidamente são as de 4 e 2 sílabas correspondendo respectivamente a 27% e 22% do total de palavras.

No caso da categoria morfológica da palavra, foi acrescentado na base de dados o atributo `Categoria` que permite armazenar a categoria da palavra.

Para se ter uma ideia da distribuição das palavras pelas categorias usadas inicialmente, a Tabela 11 mostra o número de palavras existente em cada uma das categorias.

Para o sistema utilizar os modelos estatísticos de língua na sugestão de palavras, foram acrescentadas as seguintes classes no léxico:

- 2Gram - para armazenar os *bigramas* do modelo de língua;
- 3Gram - para armazenar os *trigramas* do modelo de língua;
- 4Gram - para armazenar os *quadrigramas* do modelo de língua.

O resultado da classificação de poemas é determinístico e apenas depende do facto de existirem ou não as palavras que o constituem no léxico. A forma como é feita a classificação, quando não existe uma palavra no léxico, depende da posição da palavra em falta. Se a palavra se

encontrar no meio de um verso, então apenas o número de sílabas desse verso não estará correcto. Se a palavra em falta corresponder a uma palavra final do verso, então também o resultado de classificação da rima não estará correcto.

No caso da sugestão de palavras, não existe este problema pois, se não existir uma palavra no léxico, o sistema nunca poderá sugerir essa palavra. Apenas a utilização das aplicações externas que geram as classes possíveis não é compatível com as classes apresentadas. Foi então necessário compatibilizar as classes geradas pela aplicação *smorph* [Ait-Mokhtar, 1998]com as classes apresentadas.

Categoria	Descrição	Nº Palavras
v	verbo	4168
nc	nome comum	3565
adj	adjectivo	1692
np	nome próprio	360
adv	adverbio	119
cp	contracção pronominal	25
pind	pronome indefinido	14
in	interjeição	13
nord	numeral ordinal	7
nn	letra ou símbolo	6
ppes	pronome pessoal	6
con	conjunção	6
card	número cardinal	5
prep	preposição	4
prel	pronome relativo	4
pdem	pronome demonstrativo	4
pint	pronome interrogativo	1
art	artigo	1
ppos	pronome possessivo	1
pref	prefixo (ex-)	0
suf	sufixo (-se)	0
sem classe	sem classe	1504

Tabela 11 – Contagem das palavras com igual categoria.

4.5 Exemplos de Classificação de Poemas

Seguidamente apresentam-se alguns exemplos de classificação que demonstram o funcionamento do protótipo construído.

Em primeiro lugar apresentam-se na Tabela 12 duas quadras populares de António Aleixo e o respectivo resultado de classificação. Embora as quadras sejam de 7 sílabas métricas o resultado está apresentado em sílabas gramaticais. O resultado divide-se em duas partes. Na primeira apresenta-se o resumo de classificação do poema e a classificação das estrofes do poema e na segunda apresenta-se um exemplo de detalhe de classificação dos versos da primeira estrofe.

Quadras Populares	Classificação
Eu não tenho vistas largas,	Classificação do Poema
Nem grande sabedoria,	Resumo
Mas dão-me as horas amargas	linhas: 9 ; versos: 8 ; estrofes: 2
Lições de Filosofia.	sílabas: [8,8,8,8,0,8,9,8,8]
	rimas: [A,B,A,B, ,C,D,C,D]
Há luta por mil doutrinas.	Classificação das estrofes
Se querem que o mundo ande,	1ª estrofe - quadra [4 versos] - rima cruzada
Façam das mil pequeninas	2ª estrofe - quadra [4 versos] - rima cruzada
Uma só doutrina grande.	

Tabela 12 – Classificação de quadras populares de António Aleixo.

A informação de classificação que se obtém inclui:

- O número total de linhas do poema;
- O número total de versos do poema;
- O número total de estrofes;
- O número de sílabas de cada verso, sendo as linhas de separação das estrofes assinaladas com o número 0;
- A disposição estrófica da rima assinalada com as letras respectivas.

- Por cada estrofe é ainda apresentada a classificação quanto ao número de versos e quanto à rima.

A Tabela 13 apresenta a segunda parte da classificação que corresponde ao detalhe de classificação dos versos da primeira estrofe. O detalhe de classificação acrescenta à informação anterior a designação da classificação das estrofes quanto ao número de sílabas e quanto ao tipo da última palavra.

Detalhe de Classificação
1ª estrofe - quadra [4 versos] - rima cruzada
1º Verso - octossílabo [8 sílabas] - grave - A
2º Verso - octossílabo [8 sílabas] - grave - B
3º Verso - octossílabo [8 sílabas] - grave - A
4º Verso - octossílabo [8 sílabas] - grave - B

Tabela 13 – Detalhe de classificação da primeira quadra de António Aleixo.

O segundo exemplo é uma estrofe dos Lusíadas como mostra a Tabela 14. Embora este exemplo seja uma oitava-rima com versos de 10 sílabas métricas, o resultado está apresentado em termos de sílabas gramaticais. Pode-se ver que a estrofe é composta por 8 versos, e, tal como no exemplo anterior, o número versos por estrofe são constantes. Também a rima obedece a um formato rígido do tipo (ABABABCC).

Estrofe dos Lusíadas	Classificação
As armas e os barões assinalados	Classificação do Poema
Que da ocidental praia lusitana	Resumo
Por mares nunca de antes navegados	linhas: 8 ; versos: 8 ; estrofes: 1
Passaram ainda além da Taprobana,	sílabas: [12,12,12,13,12,13,13,11]
Em perigos e guerras esforçados	rimas: [A,B,A,B,A,B,C,C]
Mais do que prometia a força humana,	Classificação por estrofes
E entre gente remota edificaram	1ª estrofe - oitava [8 versos] - rima cruzada,
Novo Reino, que tanto sublimaram;	rima emparelhada.

Tabela 14 – Classificação de uma estrofe dos Lusíadas.

Destaca-se ainda a utilização de diferentes tipos de disposição estrófica da rima na mesma estrofe, ou seja, a estrofe apresenta rima cruzada nos quatro primeiros versos e rima emparelhada nos dois últimos versos.

O detalhe de classificação desta estrofe também corresponde a versos de 12 sílabas em que se aplicou a regra geral para a sua designação '12 sílabas'. Quanto ao tipo da última palavra todos os versos apresentam uma palavra grave.

O terceiro exemplo, descrito na Tabela 15, corresponde a duas estrofes realizadas por crianças em idade escolar. É de salientar o facto dos versos realizados por crianças não apresentarem uma grande regularidade quer em termos de número de versos quer em termos de número de sílabas. O mesmo já não se pode dizer em relação à rima.

Estrofes de Crianças	Classificação
menina que leva a vida	Classificação do Poema
sentadinha a escrever,	Resumo
faça favor de ensinar,	linhas: 8 ; versos: 7 ; estrofes: 2
eu também quero aprender.	sílabas: [9,8,8,8,0,4,5,4]
	rimas: [A,B,C,B, ,C,C,C]
Brincar, brincar	Classificação por estrofes
és para brincar	1ª estrofe - quadra [4 versos] - rima cruzada
e alegrar	2ª estrofe - terceto [3 versos] - rima seguida

Tabela 15 – Classificação de estrofes realizadas por crianças

Neste exemplo, pode-se ver que as estrofes não têm o mesmo número de versos, sendo a primeira estrofe composta por quatro versos, e a segunda por três versos. Também o número de sílabas de cada verso varia de verso para verso. No caso da rima, tem um formato diferente em cada estrofe.

4.6 Exemplos de Sugestão de Palavras

Na sugestão de palavras existem várias hipóteses possíveis que condicionam a forma como são seleccionadas as palavras. Seguidamente apresentam-se quatro situações diferentes para sugestão de palavras a partir da quadra popular de António Aleixo:

“*A Quem prende a água que corre*

É por si próprio enganado;

O ribeirinho não morre,

Vai correr por outro lado.”

Supondo que se pretende sugerir a última palavra da quadra: ‘lado’, e tendo já introduzido os versos anteriores, no primeiro exemplo apresenta-se uma situação em que não foi definida nenhuma estrutura de poema nem existe nenhuma quadra completa de onde se possa inferir qual a rima ou número de sílabas. Também não são tidas em conta as palavras escritas anteriores do último verso. Este é o caso mais desfavorável pois as palavras sugeridas apenas têm em conta a frequência de ocorrência e, por isso, o resultado das primeiras vinte palavras apresentado na Tabela 16 mais parece um conjunto de palavras que nada têm a ver com o poema.

Primeiras 20 Palavras Sugeridas
de, a, e, o, que, do, da, um, em, para, os, uma, não, com, é, no, por, na, as, dos, ...

Tabela 16 – Sugestão de palavras por frequência de ocorrência.

Em segundo lugar apresenta-se, a contabilização da palavra que antecede a palavra que se pretende sugerir que neste caso é ‘outro’. Esta situação tem em conta os *bigramas* do modelo de língua e seleccionará apenas aqueles que têm como primeira palavra a palavra ‘outro’ e ordenará o resultado por frequência. Como se pode ver no resultado da Tabela 17, embora apareça em primeiro lugar a palavra que se pretendia, existem ainda muitas que parecem fora do contexto.

Primeiras 20 Palavras Sugeridas
lado, dos, a, de, que, o, para, e, dia, em, aspecto, é, com, no, mundo, jogador, não, do, caso, os, ...

Tabela 17 – Sugestão de palavras por frequência de ocorrência de pares de palavras.

Em terceiro lugar apresenta-se a situação em que apenas está definida a rima da palavra a sugerir. No caso da quadra apresentada, como a rima é da forma ABAB e o segundo verso termina com a palavra ‘enganado’ então a sugestão é composta das palavras que têm a mesma rima e ordenadas por frequência de ocorrência. A Tabela 18 mostra as primeiras dez palavras.

Primeiras 10 Palavras Sugeridas
lado, passado, resultado, dado, deputado, avançado, machado, demasiado, obrigado, advogado, ...

Tabela 18 – Sugestão de palavras por rima.

Em quarto lugar apresenta-se a situação em que está definido o número de sílabas de cada verso. Para o exemplo do último verso da quadra apresentada, e supondo que se pretendiam versos com oito sílabas então o número de sílabas anteriores à palavra que se pretende sugerir contabilizam 6 sílabas pelo que se pretende palavras com apenas 2 sílabas. A Tabela 19 mostra as primeiras dez palavras que satisfazem esta condição.

Primeiras 10 Palavras Sugeridas
para, uma, como, pelo, também, sua, pela, está, anos, entre, ...

Tabela 19 – Sugestão de palavras por número de sílabas.

Em quinto lugar apresenta-se a situação mais favorável em que estão definidas a rima e o número de sílabas da palavra a sugerir. Neste caso obtém-se o resultado apresentado na Tabela 20 que contém as palavras existentes no léxico que satisfazem ambas as condições.

Palavras Sugeridas
lado, dado, fado, gado, prado, grado

Tabela 20 – Sugestão de palavras por rima e por número de sílabas.

Por último, apresenta-se um exemplo em que para além das restrições de rima e de número de sílabas se usa a palavra anterior para filtrar as palavras a sugerir. Neste caso apenas são sugeridas 3 palavras.

Palavras Sugeridas
lado, dado, fado

Tabela 21 – Sugestão de palavras por rima e por número de sílabas com palavra anterior.

5 IMPLEMENTAÇÃO DO SISTEMA LUCAS

5.1 Interface do Sistema

Foi elaborada uma versão autónoma do sistema cuja interface é apresentada na Figura 17.

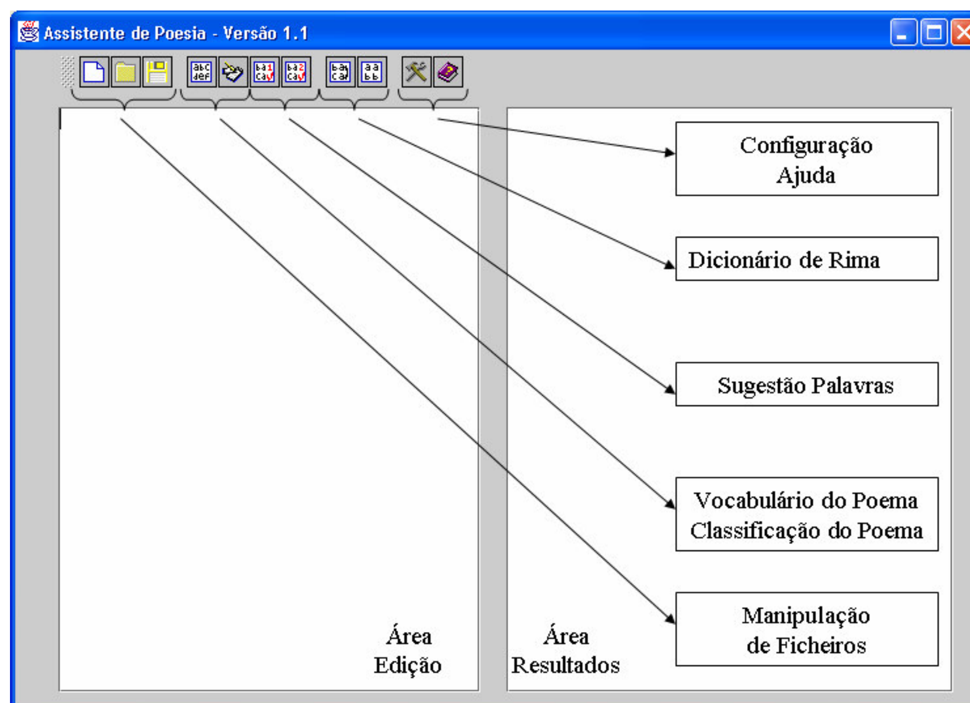


Figura 17 – Interface do Sistema *LuCas*.

A interface do sistema é composta por 2 áreas de texto. A área que se apresenta do lado esquerdo na Figura 17 corresponde à área de edição dos poemas onde o utilizador tem disponíveis os comandos de edição de um editor de texto. A área que se apresenta do lado direito corresponde

à área de resultados onde são apresentados os resultados correspondentes aos comandos efectuados pelo utilizador.

Os botões que se apresentam por cima das áreas de texto correspondem aos comandos disponíveis estando agrupados por funcionalidades. A Tabela 22 contém uma explicação sumária dos vários comandos sendo o primeiro grupo de comandos as funções de manipulação de ficheiros, o segundo grupo as funções de classificação do poema e sugestão de palavras, o terceiro grupo contém as funções de um dicionário de rima e o quarto grupo contém as funções de configuração e ajuda.

	Novo Poema – Limpa a área de edição do poema; Abrir Poema – Abre um ficheiro que contenha o poema; Gravar Poema – Grava o conteúdo da área de edição num ficheiro;
	Verificar Vocabulário – Verifica se as palavras existem no léxico; Classificar Poema – Classifica o poema; Sugestão 1 – Sugere palavras usando o poema já introduzido. Sugestão 2 – Sugere palavras usando a configuração definida.
	Palavras que Rimam – Devolve as 1as N palavras que rimam. Agrupar por Rima – Agrupa as palavras por rima.
	Configurações – Permite configurar o sistema. Ajuda – Ecrã de ajuda do sistema.

Tabela 22 – Comandos do sistema *Lucas*.

5.2 Avaliação do Classificador de Poemas

A avaliação do classificador de poemas foi feita com base num conjunto de poemas maioritariamente realizadas por crianças em idade escolar. Foram utilizadas quadras de António Aleixo e dos Lusíadas. O conjunto de poemas de teste é composto por cerca de duas centenas de estrofes de crianças, uma dezena de quadras de António Aleixo e as primeiras duas estrofes dos Lusíadas. Foram realizados testes com cerca de 25 estrofes, das quais, a maior parte realizadas por crianças, e algumas de António Aleixo e Camões.

Estes poemas não foram usados na criação dos modelos de língua, usados na sugestão de palavras, no entanto foi necessário verificar se todas as palavras existentes nos poemas existiam no dicionário de palavras. As palavras que não existiam foram inseridas bem como a sua respectiva transcrição fonética e divisão silábica.

Apenas se nota um maior tempo de classificação quando se usam poemas mais longos compostos por várias estrofes. Uma das razões deve-se ao facto de que a verificação de rima é feita para todo o poema , ou seja, quando se atribui uma letra 'A' para uma determinada rima, todos os versos do poema que terminarem com a mesma rima irão ter a mesma letra 'A' para se poder verificar qual a regularidade da rima.

Também a contagem silábica dos versos do poema contribui para a degradação do tempo de classificação dos poemas, pois é necessário contabilizar a divisão silábica de todas as palavras que compõem o poema.

Para se ter uma ideia dos tempos de resposta associados, foi realizada uma experiência com vários poemas de diferentes dimensões em que foram contabilizados os tempos de resposta do sistema. Dos vários testes realizados foram seleccionados alguns dos valores obtidos que estão resumidos na Tabela 23.

N.º de Estrofes	N.º de Palavras	Tempo de Resposta [s]
1	7	2
2	23	9
1	54	13
2	29	13
2	35	11
3	57	20
4	77	30
5	107	44
6	131	59

7	150	78
9	236	156

Tabela 23 – Tempos de Classificação.

Para analisar os resultados, foi feito o gráfico da Figura 18 com os valores correspondentes aos tempos obtidos em que se colocou no eixo das abcissas o número de palavras e no eixo das ordenadas o tempo em segundos.

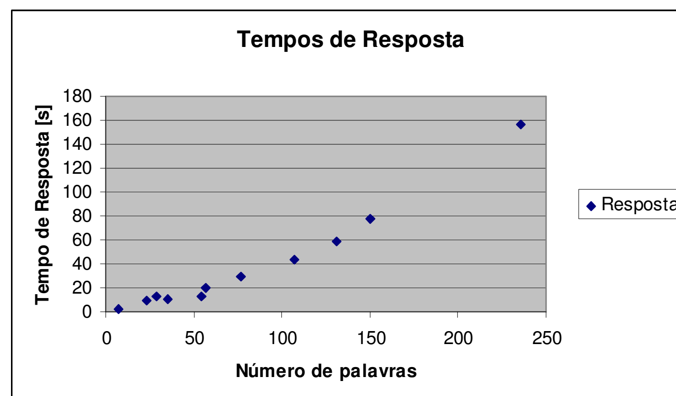


Figura 18 – Gráfico dos Tempos de Resposta de Classificação.

Como se pode observar na Figura 18, o tempo de resposta aumenta com o aumento do número de palavras que constituem o poema. Este resultado já era esperado uma vez que quanto maior for o número de palavras mais tempo processamento é necessário para contabilizar o número de sílabas dos versos, a rima do poema e as regras de classificação a aplicar ao poema.

A opção de partilhar uma estrutura de dados pelos três módulos evita ter de andar a copiar a informação pelos vários módulos o que provocaria um tempo de demora significativamente maior que o resultado obtido. Por outro lado esta opção aumenta o acoplamento entre os módulos restringindo a generalidade e independência destes módulos, ficando esta implementação comprometida com a estrutura de dados usada.

5.3 Avaliação do Preditor de Palavras

A primeira avaliação teve como objectivo verificar qual a diferença de utilizar os grupos de 2, 3 e 4 palavras dos modelos de língua descritos, para sugestão de palavras finais dos versos, sem

ter em conta as restrições estruturais de número de sílabas e rima. Como metodologia foram seguidos os seguintes passos:

1) Em primeiro lugar selecciona-se a primeira estrofe e o primeiro verso;
2) Em segundo lugar remove-se a última palavra do verso seleccionado;
3) Em terceiro lugar selecciona-se, caso exista, a palavra anterior à palavra removida (*PA*). Utilizando os grupos de 2 palavras, ordenam-se por ordem decrescente de frequência de ocorrência todos os pares de palavras em que a primeira palavra do par seja igual a *PA*. Para efeitos de registo apenas se guardam as primeiras 10 palavras do resultado;
4) Em quarto lugar seleccionam-se, caso existam, as duas palavras anteriores à palavra removida (*PA1* e *PA2*). Utilizando os grupos de 3 palavras, ordenam-se por ordem decrescente de frequência de ocorrência todos os trios de palavras em que as primeiras duas palavras do trio sejam iguais a *PA1* e *PA2*. Para efeitos de registo apenas se guardam as primeiras 10 palavras do resultado;
5) Em quinto lugar seleccionam-se, caso existam, as três palavras anteriores à palavra removida (*PA1*, *PA2* e *PA3*). Utilizando os grupos de 4 palavras, ordenam-se por ordem decrescente de frequência de ocorrência todos os grupos de quatro palavras em que as primeiras três palavras do trio sejam iguais a *PA1*, *PA2* e *PA3*. Para efeitos de registo apenas se guardam as primeiras 10 palavras do resultado;
6) Em sexto lugar selecciona-se o próximo verso da estrofe e continua-se no passo 2.

Este processo de teste apresenta como resultado o conjunto das palavras que, de acordo com o modelo, têm maior probabilidade de ocorrência tendo em conta 1, 2 ou 3 palavras anteriores para a última palavra de cada verso que constitui a estrofe em teste.

Os resultados obtidos foram divididos por grupos de palavras:

- Para o grupo de 4 palavras, na maioria dos casos, não existem conjuntos de 4 palavras que contenha as 3 primeiras palavras anteriores à palavra a sugerir. Nos casos em que existem resultados, não atingem as 10 hipóteses;
- Para os grupos de 3 palavras, na maioria dos casos, existem soluções com 10 hipóteses em que cerca de metade poderiam ser palavras passíveis de ser usadas no verso. Existem casos em que o resultado não atinge as 10 hipóteses e alguns em que não existe nenhuma solução;
- Para os grupos de 2 palavras, não foi encontrado nenhum caso sem o total das 10 soluções. No entanto, o número de palavras que seriam possíveis de ser usadas também é cerca de metade como nos grupos de 3 palavras. Neste caso, quase sempre

existe uma hipótese mesmo que não seja nas 10 primeiras em que a palavra que foi removida está contemplada.

Estes resultados permitem concluir que, na sugestão de palavras, os *bigramas* são os grupos de palavras que permitem, na maioria dos casos, obter a palavra que se pretende sugerir.

A segunda avaliação teve como objectivo verificar qual a diferença, na sugestão de palavras, quando se entra em conta com as restrições estruturais de número de sílabas e de rima nos poemas utilizando os mesmos poemas de teste da avaliação anterior e o mesmo processo de remoção da última palavra de cada verso.

Como metodologia seguida foram seguidos os seguintes passos:

1) Em primeiro lugar apenas foram consideradas as restrições de número de sílabas das palavras ordenadas por frequência de ocorrência;
2) Em segundo lugar apenas foram consideradas as restrições de rima das palavras ordenadas por frequência de ocorrência;
3) Em terceiro lugar foram consideradas as restrições de número de sílabas em conjunto com a rima e ordenadas por frequência de ocorrência.

Os resultados obtidos foram divididos em três grupos:

- Utilizando como restrição apenas o número de sílabas das palavras, os resultados permitem concluir que nas 10 primeiras palavras sugeridas existe maior número de palavras que podiam ser usadas em substituição da palavra removida comparativamente aos modelos de língua de *bigramas*;
- Utilizando como restrição apenas a rima das palavras, também neste caso, os resultados permitem concluir que nas 10 primeiras palavras sugeridas existe maior número de palavras que podiam ser usadas em substituição da palavra removida comparativamente aos modelos de língua de *bigramas*;
- Utilizando como restrição o número de sílabas das palavras em conjunto com a rima das palavras, o número total de palavras sugeridas reduz consideravelmente e a maior parte das palavras podia ser usada em substituição da palavra removida. Esta opção corresponde à melhor opção para filtrar as palavras a sugerir. O único inconveniente é que nem sempre existe a informação que a permite aplicar.

Pode-se concluir que o melhor resultado de sugestão de palavras é obtido a partir da conjunção das várias abordagens seguidas para filtrar as palavras. Das várias hipóteses individuais aquela que obtém melhores resultados é a rima seguida do número de sílabas das palavras. São as duas em conjunto que obtém o melhor resultado de sugestão de palavras.

As avaliações quer do módulo de classificação quer do módulo de predição não confrontam com outras ferramentas enumeradas pois nas ferramentas descritas, não existe nenhuma que seja comparável em funcionalidade ao sistema proposto.

6 CONCLUSÕES

6.1 Situação Actual

Para fazer um ponto de situação do trabalho realizado, é necessário ter em conta os objectivos iniciais que serviram como linha orientadora para a implementação das funcionalidades e para os resultados dos testes realizados.

Globalmente, pode-se dizer que os objectivos iniciais foram atingidos, ou seja, as funcionalidades que se pretendiam implementar foram, em alguns casos, completamente resolvidas e, noutros casos, embora não completamente automatizadas conseguiram-se, mesmo assim, realizar testes no sentido de validar as ideias aqui expressas.

Conceptualmente, podem-se dividir os objectivos iniciais em dois grupos: por um lado, os objectivos relacionados com a classificação de poemas e, por outro lado, os objectivos relacionados com a predição das palavras.

Começando pelo classificador de poemas, um dos aspectos importantes a realçar é a sua arquitectura. Por um lado, permite uma grande flexibilidade e adaptação, mesmo quando se testavam as várias abordagens, devido à sua natureza modular. Por outro lado, se se olhar para os testes de resposta efectuados, verifica-se que os aspectos de eficiência têm de ser melhorados quando se pretendem classificar poemas mais extensos.

O tempo de classificação dos poemas é um factor importante a ter em conta. Mas também é um facto que, se se aumentar o poder computacional, obtém-se uma significativa melhoria no tempo de classificação quando se tratam poemas de maior dimensão.

O facto de se optar por uma organização em módulos, contribui para que o sistema perca eficiência, contudo as vantagens que traz em termos de versatilidade de utilização do sistema e adaptação às alterações necessárias para teste das várias abordagens compensam essa perda de desempenho.

O facto de não realizar a integração deste sistema no GalInHa [Matos et Al., 2003] não permite explorar completamente as aplicações externas utilizadas. Durante a construção do protótipo houve a necessidade de arranjar um mecanismo que permita guardar a informação gerada por essas aplicações. Esta solução limita os resultados obtidos e corta hipóteses de soluções que, no caso de ter as aplicações disponíveis, não sucede.

A solução encontrada com a implementação da base de dados permite também guardar a informação gerada pelas aplicações externas. Mas se por um lado esta opção permite ter o sistema a funcionar autonomamente, mesmo quando não estão disponíveis as aplicações externas, por outro lado surge o problema quando aparece uma palavra que não existe no léxico. Se as aplicações externas estivessem disponíveis conseguiriam gerar a informação relativa a essa palavra mas não estando disponíveis no léxico, a palavra é considerada como não válida.

O tratamento das palavras que não existem no léxico tem diferentes consequências para as diferentes funcionalidades do sistema e dependendo da sua posição nos versos.

Se as palavras em falta estão no final dos versos então pelo facto de não ser possível obter a informação de número de sílabas e a informação da rima destas palavras o resultado de classificação é incompleto e a sugestão de palavras não considera a rima como critério de selecção.

Se as palavras em falta estão no meio dos versos então o resultado de classificação é incompleto com a informação de classificação relativa a rima do poema correcta. Neste caso é possível considerar a rima como critério de selecção na sugestão de palavras.

Na verificação da rima entre as palavras as opções de implementação mostram que a abordagem seguida, tendo como comparação os dicionários de rima em formato electrónico que foram testados, permite realizar a verificação da rima das palavras de uma forma fiável, pois tem como base a transcrição fonética das palavras em alternativa à terminação em termos de letras das palavras.

Também o facto de existirem várias formas diferentes de dizer a mesma palavra origina a que exista várias transcrições fonéticas para a mesma palavra. Surgiu a assim a necessidade de arranjar um critério de escolha para a transcrição fonética a utilizar.

Esta escolha tem maior impacto quando as palavras sugeridas têm poucas alternativas de escolha e por isso nestes casos são contempladas as várias transcrições fonéticas. Nos casos em que o número de palavras sugeridas é elevado, é escolhida apenas uma das transcrições fonéticas.

Quanto à utilização deste sistema, houve a preocupação de implementar uma interface simples que seja fácil de usar por quem inicia o estudo de poesia, como é o caso das crianças em idade escolar. Mas para que o sistema fique vocacionado para este fim ainda muito há a fazer em

termos de robustez e ecrãs de ajuda. Também a interface pode ser melhorada no sentido de tornar a utilização do sistema mais fácil como é exemplo a existência de teclas de atalho.

Para poetas este sistema permite ajudar a escolher as regras da poética portuguesa a colocar no poema e fazer para a poesia um pouco como os correctores ortográficos fazem para os textos em prosa, pois como já foi referido, neste trabalho não houve o objectivo de criar poesia automaticamente.

Um aspecto que convém salientar é que, mesmo classificando diferentes tipos de poemas, devido às restrições impostas nas definições, existem poemas que não estão contemplados neste sistema, como é o caso dos poemas modernos em que as regras da poética portuguesa não são cumpridas, dando liberdade ao poeta para uma poesia com menos imposições.

Focando agora os aspectos relacionados com o preditor de palavras, os processos, por vezes manuais, de teste de algumas abordagens de selecção e ordenação das palavras a sugerir condicionaram os resultados obtidos.

A predição das palavras está condicionada pela existência das palavras no léxico. Isto significa que mesmo com ligação às aplicações externas a selecção das palavras é feita apenas sobre as palavras existentes no léxico.

A utilização das categorias das palavras, para realizar a selecção de palavras, não reduz significativamente o número de palavras a sugerir. Por um lado deve-se ao facto do número de categorias possível por cada sugestão ter muitas categorias possíveis e por outro lado devido ao facto das restrições estruturais apresentarem uma grande diferença em relação a todas as outras alternativas.

No caso dos modelos de língua utilizados, também o processo manual de teste limitou muito os testes realizados. Talvez a utilização de diferentes modelos, referentes a diferentes corpus de texto possa alterar os resultados obtidos.

6.2 Perspectivas Futuras

São várias as portas que ficaram abertas e as soluções que podem ser melhoradas em trabalhos futuros.

A disponibilização deste sistema no GalInHa [Matos et Al., 2003], é uma aspecto que permitirá confirmar algumas das suposições aqui levantadas e melhorar os resultados obtidos relativamente às experiências de teste que foram realizadas com processos manuais.

A interface Web disponibiliza de forma fácil através de um browser o acesso e utilização das várias aplicações e respectivas funcionalidades que aí estão disponíveis. Espera-se que as preocupações de implementação de forma modular facilitem a integração das diferentes funcionalidades do sistema proposto na interface comum a todas as aplicações.

Com a integração deste trabalho espera-se ainda:

- O acesso mais simples à informação gerada pelas aplicações externas;
- A disponibilização das funcionalidades do sistema para outras aplicações;

Outro aspecto a salientar é que o facto de integrar o sistema na plataforma descrita, permite dispensar algumas das classes e das funcionalidades que apenas foram acrescentadas no léxico para tornar possível o funcionamento autónomo do sistema.

Também na arquitectura do sistema, uma das melhorias que poderá ser implementada, é realizar uma linguagem de comandos que permita activar as diversas funcionalidades dos diversos módulos de uma forma mais flexível e dinâmica. Desta forma a interface com o sistema fica mais bem definida e independente da implementação das funções internas de classificação e de sugestão de palavras.

A alteração, acréscimo ou remoção de funcionalidades dos módulos obriga, neste momento, a uma alteração dos processos de coordenação para contemplar essas alterações. Com a possibilidade de activação independentemente das funcionalidades poder-se-á também melhorar a facilidade de alteração e configuração do sistema.

Para resolver os aspectos de desempenho do sistema, será necessário construir uma versão diferente do sistema em que se coloca em segundo plano as questões de flexibilidade de alteração e adaptação, e se canaliza todo o esforço para realização de código optimizado. Também a linguagem poderá ser escolhida por forma a tirar melhor partido da máquina. E por fim a escolha de uma máquina mais rápida dará, com certeza, uma ajuda adicional.

Relativamente ao ritmo dos versos, existem alguns aspectos que podem ser mais aprofundados e investigados como são o caso dos acentos predominantes, cuja distribuição ao longo do verso provoca alteração no seu ritmo, isto é, na sua musicalidade e cadência. Para além do acento tónico da última sílaba métrica, há outros cuja colocação varia com o metro ou medida dos versos.

Em relação à divisão silábica dos versos, também aqui se pode aprofundar um pouco mais as a implementação da detecção das sílabas gramaticais para contemplar junção entre vogais de palavras.

Para se conseguir este objectivo terão de ser implementadas as regras descritas anteriormente que caracterizam as formas de contracção e de diérese.

Os dois últimos melhoramentos propostos permitem inclusivamente fornecer a informação necessária para que um mecanismo automático de leitura em voz alta de poemas passe a contemplar a musicalidade e cadência do poema para assim melhorar o ritmo de leitura.

As funcionalidades associadas à rima que foram executadas no contexto deste trabalho poderão ser complementadas de modo a integrar no sistema todas as funcionalidades de um dicionário de rimas, com a vantagem da abordagem seguida ser com base nas transcrições fonéticas das palavras. A interface do sistema terá de sofrer alterações para disponibilizar estas funções.

Relativamente ao preditor de palavras, um dos aspectos que, sem dúvida, pode melhorar, é a utilização das categorias das palavras para filtrar ainda mais as palavras a sugerir. Para se conseguir resultados mais precisos será necessário usar modelos estatísticos de categorias de palavras ou associar factores de confiança às categorias possíveis para a próxima palavra.

A influência de utilização de diferentes modelos de língua na sugestão de palavras é um aspecto que fica em aberto, uma vez que apenas foram realizadas experiências com um modelo.

Para se conseguir ter vários modelos de língua será necessário arranjar diferentes corpus de texto e para cada um gerar o respectivo modelo de língua.

REFERENCIAS

[Ait-Mokhtar, 1998] - Salah Ait-Moktar, "L'analyse présyntaxique en une seule étape", Tese de Doutoramento, Université Blaise Pascal, Clermont-Ferrand, França, 1998.

[Araújo & Mamede, 2002] - Paulo Araújo, Nuno Mamede, "Classificador de Poemas", CCTE 2002, Lisboa, Portugal, Maio 2002.

[Areal, 2000] - Américo Areal, "Curso de Português", Edições ASA 15ª Edição, 2000.

[Batista, 2002] - Fernando Batista, "Análise Sintáctica de Superfície e Consistência de Regras", Tese de Mestrado, Instituto Superior Técnico, Lisboa, Portugal, (Trabalho em Curso).

[Chachanashvili, 1991] - Alex Chachanashvili, "Dada Poem Generator", 1991, http://www.achacha.org/cgi-bin/dada.cgi.

[Chen et al., 1998] - Chen, S., Beeferman, D., Rosenfeld, R., "Evaluation Metrics for Language Models.", In Proc. DARPA Broadcast News Transcription and Understanding Workshop (BNTUW), Lansdowne, Virginia, February 1998.

[Chomsky, 1965] - Chomsky, N., "Aspects of the Theory of Syntax", Cambridge, MIT Press, 1965.

[Clarkson & Rosenfeld, 1997] - Clarkson P., Rosenfeld, R., "Statistical Language Modeling using the CMU-Cambridge toolkit", In Proc. Eurospeech '97, September 1997.

[Coelho, 1987] - Jacinto Prado Coelho, "Dicionário de Literatura", Editora Minho 3ª Edição, 1987.

[Cohen, 2001] - Harold Cohen, "AARON the Cybernetic Artist", Kurzweil CyberArt Technologies, 2001, http://www.kurzweilcyberart.com/.

[E-Poetry] - "Electronic Poetry Center", http://epc.buffalo.edu/e-poetry/

[Early, 1970] - Early, J., "An efficient context-free parsing algorithm", 1970.

[Faiza, 1999] - Abbaci Faiza, "Développement du Module Post-SMorph", Tese de Mestrado, Université Blaise Pascal, Clermont-Ferrand, França, 1999.

[Ferreira et al., 2001] - Nuno Ferreira, Joana Paulo, Ana Pacheco, "O Poeta", Relatório do Projecto de Introdução aos Agentes Autónomos, Lisboa, Portugal, 2001.

[Frykholm, 1996] - Niklas Frykholm, "Electric Poet", 1996,

http://www.macinsearch.com/infomac/game/word/mcpoet-43.html.

[Garcia & Oliveira, 2001] - Luís Garcia, Luís Oliveira, “Eugénio, o génio das palavras”, Predictor de Palavras para o Português Europeu, Beja e Lisboa, Portugal, 2001.

[Geração Poesia] - “Geracao Poesia”, http://www.geracaopoesia.meublog.com.br/

[Good, 1953] - Good, I. J., “The population frequencies of species and the estimation of population parameters. Biometrika, 1953.

[Hagège, 2000] - Caroline Hagège, “Analyse Syntaxique Automatique du Portugais”, Tese de Doutoramento, Université Blaise Pascal, Clermont-Ferrand, França, 2000.

[Huang et al., 2001] - Huang, Xuedong, Acero, Alex, Hon, Hsiao-Wuen, “Spoken Language Processing: A Guide to Theory, Algorithm, and System Development”, Prentice Hall, 2001

[Hutchens, 1995] - Hutchens, Jason L., “Natural Language Grammatical Inference”, “An Honours Dissertation in Information Technology”, University of Western, Australia, 1995.

[Jorge et al., 2000] - Daniela Jorge, Flávia Henriques, Cátia Jorge, Escola Básica do Sobral da Abelheira, Mafra, 2000.

[Jurafsky & Martin, 2000] – Daniel Jurafky, James H. Martin, “Speech and Language Processing - An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition, Prentice Hall, 2000”

[Katz, 1987] - Katz, S. M., “Estimation of probabilities from sparse data for the language model component of a speech recognizer”, IEEE Transactions on Acoustics, Speech and Signal Processing, 1987.

[Kurzweil, 1999] - Kurzweil, Ray, “Ray Kurzweil’s Cybernetic Poet”, Kurzweil CyberArt Technologies, 1999, http://www.kurzweilcyberart.com/poetry/rkcp_overview.php3.

[Mamede et al., 2004] - Nuno Mamede, Isabel Trancoso, Paulo Araújo, Céu Viana, “Poetry Assistant”, ICSLP 2004, Outubro de 2004, Jeju Island, Korea.

[Mateus & Graça, 2002] - Alexandre Mateus, João Graça, “Interface Web para o Sistema Galaxy Communicator”, Relatório de Trabalho Final de Curso, Lisboa, Portugal, 2002.

[Matos et Al., 2003] - Matos, D. M., Paulo, J. L., Mamede, N. J., “Managing Linguistic Resources and Tools”, Lecture Notes in Artificial Inteligence, no. 2721, Springer-Verlag, pp. 135-142, 2003.

[Matos et al., 2002] - Matos, D., Mateus, A., Graça J., Mamede, N., “Empowering the User: a Data-Oriented Application-Building Framework”, Adjunct Proceedings of the 7th ERCIM Workshop "User Interfaces for All", Paris, France, 2002, (not yet published).

[Moisés, 1974] - Massud Moisés, “Dicionário de Termos Literários”, Editora Coltrix, 1974.

[Oliveira, 1996] - Luís Oliveira, "Síntese de Fala a Partir de Texto", Tese de Doutoramento, Instituto Superior Técnico, Lisboa, Portugal, 1996.

[Paulo & Mamede, 2001] - Joana Paulo, Nuno Mamede, "PAsMo - Pós Análise Morfológica", Manual Técnico, Lisboa, Portugal, 2001.

[Poetry Library] – "The International Library of Poetry", http://www.poetry.com

[Polifroni & Seneff, 2000] - Polifroni, J., Seneff, S., "GALAXY-II as an Architecture for Spoken Dialogue Evaluation", in Proc. LREC, Athens, Greece, May 31-June 2, 2000, http://www.sls.lcs.mit.edu/sls/publications/2000/lrec-2000.pdf.

[Pretor, 2000] - Pretor Informática e Sistemas Ltda., "Dicionário de Rimas Poéticas", 2000, http://www.lemon.com.br

[Productions, 1997] - Poetry Ink Productions, "Chaos Poetry Generator", 1997, http://www.macinsearch.com/infomac2/game/word/chaos-poetry-generator-hc.html.

[Projecto Vercial] – "Projecto Vercial", http://www.ipn.pt/literatura/

[Rosenfeld, 2000] – Rosenfeld, R., "Two decades of Statistical Language Modeling: Where Do We Go From Here?", Proceedings of the IEEE, 88(8), 2000.

[SAM-PA] - Oliveira, Luís, "Alfabeto Fonético para o Dialecto Padrão do Português Europeu", http://www.l2f.inesc-id.pt/~lco/ptsam/ptsam.pdf

[Seneff et al., 1998] - Seneff, S., Hurley, E., Lau, R., Pao, C., Schmid, P. e Zue, V., "Galaxy-II: A Reference Architecture for Conversational System Development", in Proc. ICSLP '98, Sydney, Australia, 30 Nov.-4 Dec. 1998, 3:931-934, http://www.sls.lcs.mit.edu/sls/publications/1998/icslp98-galaxy.pdf.

[Seneff et al., 1999] - Seneff, S., Lau, R., Polifroni, J., "Organization, Communication, and Control in the GALAXY-II Conversational System", in Proc. Eurospeech 99, Budapest, Hungary, September 1999, http://www.sls.lcs.mit.edu/sls/publications/1999/eurospeech99-seneff.pdf.

[Westbury, 1997] - Chris Westbury, "Mc Poet", 1997, http://www.macinsearch.com/infomac/game/word/mcpoet-43.html.

[Witten, 1991] - Witten, I. H., Bell, T. C., "The zero frequency problem: Estimating the probabilities of novel events in adaptative text compression", IEEE Transactions on Information Theory, 1991.